

**Faculty of Science and Engineering
School of Earth and Planetary Sciences
Discipline of Spatial Sciences**

Building Footprint Extraction from LiDAR Data and Imagery Information

Yousif Abdul-kadhim Mousa

**This thesis is presented for the Degree of
Doctor of Philosophy
of
Curtin University**

June 2020

Declaration

To the best of my knowledge and belief this thesis contains no material previously published by any other person except where due acknowledgment has been made. This thesis contains no material which has been accepted for the award of any other degree or diploma in any university.

Signature:

Date: 18/06/2020

Dedication

This work dedicated to God, to the spirit of my father, mother, wife, and all family and friends. Additionally, I dedicate this work to the great Iraqi people who sacrificed themselves to keep the country safe and stable.

It is with humbleness that this contribution is made to the field of Photogrammetry and Remote Sensing.

Abstract

Automating the process of building detection and outline regularisation is an active research topic in the field of photogrammetry and remote sensing. This is due to the need for large sets of accurate and detailed data in several applications such as generation and updating building maps and databases, urban planning, and urban monitoring. The quick and continuous development of sensor technologies including laser scanning, airborne and satellite imagery introduced big challenges in the automation process together with achieving the desired geometric accuracy of the building outlines. Although significant progress has been achieved through ongoing research, the automation level and accuracy is still limited by many factors such as complexity of building shape, separation of buildings from trees, especially those located in shadow regions. The ability to effectively address these factors is dependent on the success of the applied mathematical model for rectilinear and non-rectilinear building outlines.

A comprehensive workflow for this task is proposed in this thesis. This workflow starts with detecting and cleaning outliers from laser-scanned point cloud data, then transforming this data into a regular gridded digital surface model (DSM). The digital terrain model (DTM) is extracted using a new algorithm. This algorithm utilises a generation of seed or network of ground points selected as minima within well-distributed scanlines moving across the whole DSM image. The normalised DSM (nDSM) (containing above ground objects i.e. buildings and trees) is created by subtracting DTM from DSM. In order to separate building segments from trees, the nDSM, planarity measure and the Normalized Difference Vegetation Index (NDVI) are fused in a new approach. Next, boundary points for each single segment are identified to be used in the building outline generalisation and regularisation.

For the generalisation task, an automatic procedure is proposed utilising a combination of data- and model-driven approaches to provide a robust solution. The core part of the method includes a novel data-driven approach to generate approximate building polygons from a list of given boundary points. The algorithm iteratively calculates and

stores likelihood values between an arbitrary starting boundary point and each of the following boundary points using a function derived from the geometrical properties of a building. The generated approximate building polygons usually have irregular shape and do not satisfy a best-fit with the input boundary points. Thus, a regularisation step is required.

The regularisation aims to create regular polygons (i.e. rectangular or rectilinear) associated with achieving a best fit between each edge (in the polygon) and its corresponding boundary points. This step is largely solved in this thesis through the simultaneous implementation of constraints enforcing the orthogonality of all adjacent edges in the polygon and the best fit with the input boundary point using the Gauss-Helmert Model adjustment. Non-rectilinear building polygons are modified using the Gauss-Markov Model adjustment. In the iterated least squares adjustment, updating the label/assignment of the boundary points to their correlated edges is required for further accuracy achievement. In this context, a novel labelling algorithm is proposed. The algorithm considers measuring and analysing distances and angles between the boundary points and the polygon edges. This labelling algorithm led to the proposing of a robust evaluation procedure to measure the absolute accuracy/similarity of corresponding polygons (i.e. extracted and its reference polygons). This evaluation procedure, denoted as Vertex to Model (V2M) evaluation, utilises the Root Mean Square Error (RMSE) as a standard topographic measure for assessment.

Algorithm evaluation results are presented for two datasets characterised by a variety of building outline complexity, data sources, and resolution. The algorithm results achieved an average RMSE of sub-meter accuracy.

Acknowledgements

I would like to express sincere gratitude and admiration to my PhD supervisors, Dr. Petra Helmholz and Dr. David Belton from Curtin University, Spatial Sciences. They provided me with valuable advice, guidance, support, and broke down problems to reach the main objectives of the study. Thanks to their constructive criticism and discussion my knowledge and communication skills have been significantly improved during my study. Next, I would also like to express sincere gratitude to my co-supervisor Dr. Dimitri Bulatov from Fraunhofer IOSB, Ettlingen, Germany, who joined the supervision committee in the third year of my PhD journey. I learnt much despite this short time of supervision.

I would also like to thank my colleague Hoang Long Nguyen for many discussions about research and tutoring problems in the Photogrammetry Lab for more than three years. I also want to thank my colleagues Dr. Richard Palmer who shared discussion with me during his attendance in the photogrammetry lab. Special thanks to Dr. Roland Perko from Joanneum Research, Austria, for his cooperation and providing his Matlab source code for DTM extraction as well as parameters setting. Thanks to all of the staff members and administration in the Spatial Sciences Discipline.

I would also like to thank the City of Melville, for providing me with the essential data to perform this research. Additionally, thanks to the German Society for Photogrammetry, Remote Sensing and Geoinformation (Deutsche Gesellschaft für Photogrammetrie, Fernerkundung und Geoinformation -DGPF) for the benchmark dataset provided online.

Lastly, I would like to express sincere gratitude to The Higher Committee for Education Development (HCED) in Iraq for granting me this scholarship.

Publications based on this thesis

Some of the work presented in this thesis has already been published by the thesis author and co-authors.

Paper “NEW DTM EXTRACTION APPROACH FROM AIRBORNE IMAGES DERIVED DSM” Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci., 2017, 42, 75-82.




Authors and full affiliations: Yousif Abdul-kadhim Mousa^{1a}, Petra Melmholz², David Belton³

¹ Department of Spatial Sciences, Curtin University, Perth, Australia

^a Department of Civil Engineering, Al-Muthanna University, Al-Muthanna, Iraq.

² Senior Lecturer, Curtin University, Perth, Australia

³ Senior Research Fellow, Curtin University, Perth, Australia

	Conception and design	ideas/manipulation/implementation	experimental test, analysis, and evaluation	Interpretation/ Discussion	Final approval
Yousif A. Mousa	✓	✓	✓	✓	✓
I acknowledge that these represent my contribution to the above research output. Signature:  15/01/2020					
Dr. Petra Helmholtz				✓	
I acknowledge that these represent my contribution to the above research output. Signature:  15/01/2020					
Dr. David Belton				✓	
I acknowledge that these represent my contribution to the above research output. Signature:  15/01/2020					

Paper “Building detection and regularisation using DSM and imagery information”.
The Photogrammetric Record, 34(165), 85-107.

Authors and full affiliations: Yousif Abdul-kadhim Mousa^{1a}, Petra Melmholz², David Belton³, and Dimitri Bulatov⁴


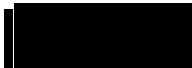
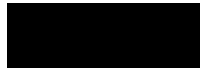

¹ Department of Spatial Sciences, Curtin University, Perth, Australia

^a Department of Civil Engineering, Al-Muthanna University, Al-Muthanna, Iraq.

² Senior Lecturer, Curtin University, Perth, Australia

³ Senior Research Fellow, Curtin University, Perth, Australia

⁴ Researcher, Fraunhofer IOSB, Ettlingen, Germany

	Conception and design	ideas/manipulation/ implementation	experimental analysis, evaluation	test, and	Interpretation/ discussion	Final approval
Yousif A. Mousa	✓	✓	✓		✓	✓
I acknowledge that these represent my contribution to the above research output. Signature:  15/01/2020						
Dr Petra Helmholz					✓	
I acknowledge that these represent my contribution to the above research output. Signature:  15/01/2020						
Dr. David Belton					✓	
I acknowledge that these represent my contribution to the above research output. Signature:  15/01/2020						
Dr. Dimitri Bulatov					✓	
I acknowledge that these represent my contribution to the above research output. Signature:  15/01/2020						

CONTENTS

Declaration	i
Dedication	ii
Abstract	iii
Acknowledgements	v
Publications based on this thesis	vi
List of figures	xi
List of tables	xix
Acronyms	xx
1 Chapter one Introduction.....	1
1.1 Building detection and boundary regularisation	1
1.2 Problem statement.....	2
1.3 Objective of the thesis	4
1.4 Significance of the research	6
1.5 Thesis organisation.....	6
2 Chapter two Background.....	8
2.1 Overview	8
2.2 Point Cloud creation.....	8
2.2.1 LiDAR.....	8
2.2.2 Imagery	10
2.3 Discussion of Lidar versus image-based point clouds	14
2.4 Data rasterisation and processing.....	15
2.4.1 Transformation of 3D point clouds to gridded DSM.....	15
2.4.2 Planarity	17
2.4.3 Mathematical Morphology.....	18
2.5 Ground Characteristics for DTM extraction	19
2.6 Mathematical model for building polygons regularisation	20
2.6.1 Gauss-Markov Model.....	21
2.6.2 Gauss-Helmert Model	22
3 Chapter three Literature review	24

3.1	Overview	24
3.2	DTM extraction	24
3.3	Building detection in terms of data sources	30
3.3.1	Building detection from 2D image data only	30
3.3.2	Building detection from 2D and 3D image data	31
3.3.3	Building detection from LiDAR data only	32
3.3.4	Fusion of LiDAR data with imagery-derived information	34
3.4	Approximate building polygons generation and refinement	36
3.4.1	Model-driven approaches	37
3.4.2	Data-driven approaches	40
3.4.3	Refinement of building polygons	43
4	Chapter four Methodology	46
4.1	Overview	46
4.2	Pre-processing	46
4.2.1	Outliers removal	48
4.2.2	Data rasterisation	49
4.3	DTM and nDSM extraction	50
4.3.1	DSM refinement	50
4.3.2	The proposed DTM extraction algorithm	51
4.4	Building segments extraction	57
4.5	Generation of approximate building polygons	59
4.5.1	Extraction of boundary points to initialise building polygon	59
4.5.2	Data-driven approach for polygon simplification	60
4.5.3	Model-driven approach for outlining rectangular buildings	64
4.6	Modelling and adjustment of building polygons	66
4.6.1	Algorithm for labelling/assigning boundary points	68
4.6.2	Refinement of non-rectilinear building polygons using GMM	74
4.6.3	Refinement of rectilinear building polygons using GHM	78
4.6.4	Initial solution and stopping criterion for the adjustment	81
4.7	RMSE as a measure for accuracy assessment of building polygons	82
5	Chapter five Evaluation	86
5.1	Study areas	86

5.1.1	Vaihingen datasets	86
5.1.2	Melville datasets.....	88
5.2	Evaluation of the proposed DTM extraction algorithm	90
5.2.1	Parameter settings for DTM extraction.....	91
5.2.2	Qualitative evaluation	93
5.2.3	Quantitative evaluation	98
5.3	Building segments extraction and evaluation	106
5.4	Evaluation of the proposed data-driven approach for building polygon generalisation	114
5.4.1	Comparison to the <i>Douglas-Peucker</i> (DP) algorithm.....	115
5.4.2	Parameters setting and analysis for building generalisation	119
5.5	Buildings Regularisation Results and evaluation.....	121
5.5.1	Quantitative evaluation of buildings regularisation	121
5.5.2	Model assumption evaluation	126
5.5.3	Assessment of the GMM adjustment	126
5.5.4	Assessment of the GHM adjustment.....	135
6	Chapter six Conclusions and discussion	143
	References	147

List of figures

Figure 2.1: Basic components of an ALS (Arefi, 2009).	9
Figure 2.2: Multiple reflection laser observation (Arefi, 2009).....	10
Figure 2.3: Generation of orthophoto images. (a) raw image. (b) rectified image provided by the German Society for Photogrammetry, Remote Sensing and Geoinformation (DGPF) (Cramer, 2010).....	14
Figure 2.4: Transformation from point clouds to gridded-DSM (Ma, 2004).....	16
Figure 2.5: Planarity measure. Orthophoto, DSM, and the planarity map from left to right.	18
Figure 2.6: Morphological operators. From left to right: binary image shows building segments, example of dilated image, and example of eroded image. Buildings are represented as white pixels; terrain is shown with black pixels.....	19
Figure 3.1: Procedure of generating RMBR. Raster image shows building segment (a). The extracted boundary points (black dots) are shown in (b). (c) shows the MBR created (red box). Non-overlapped area (black) resulting from the subtraction of the original building segment from the MBR is given in (d). (f) shows the new polygon (green) which should be extracted from the MBR. The final RMBR extracted is given in (g).....	39
Figure 3.2: Concept of the DP algorithm. (a) shows the first conducted line (blue) starts from the first vertex (red triangle) and divides the boundary points (black dots) into two sub-chains, while the red line nominates a new vertex corresponding to the farthest boundary points to the blue line. The algorithm iteratively creates straight lines from new nominated vertices that has farthest distance to the nominated edges until returning to the start point as shown in (b) and (c). (d) shows the final created polygon (green).	41
Figure 4.1: Workflow of the proposed methodology.....	47
Figure 4.2: Normal distribution of data, (a) and (b) before and after outliers removal respectively.	48
Figure 4.3: Comparison between interpolation methods. Left to right: mask representing missing data/holes, orthophoto, DSM created using linear interpolation, and nearest neighbour interpolation. The red cross indicates missing data, by Linear interpolation method partially covered this missing data	

- (DSM value = 20.4153 m), while by Nearest method, (DSM value = 18.56 m) the same as to the ground. 50
- Figure 4.4: DSM refinement: (a) before and (b) after applied median filter. (c) illustrates cross section profiles taken over a building before (red line) and after (green line) applying the median filter. 51
- Figure 4.5: Directional scanlines for minimum point selection, (a) MSD approach and (b) the new proposed Network of Ground Points (NGPs) approach. 53
- Figure 4.6: Steps for the DTM generation by the NGPs algorithm. Ground points (GPs) selected as minima on an artificial DSM (a), initial DTM (b), and vertical threshold limits (c), ground mask (d), and final DTM (e). 55
- Figure 4.7: Generation of the building mask. Orthophoto, DSM, and nDSM (across top), NDVI, Planarity Map, and Building Mask (across bottom). The red cross belongs to a tree in a shadow area. 58
- Figure 4.8: Extraction of boundary points. Left to right: building segments and the extracted boundary points. 60
- Figure 4.9: Polygonal simplification procedure from given boundary points (black dots). The first starting point (randomly selected) is shown by a black triangle, while the extracted vertices are depicted by orange triangles. The constructed lines, orthogonal distances, and polygon edges are depicted by blue, red, and green lines, respectively. Angles at the points under evaluation are depicted by pink lines. (a) shows constructed straight lines from the start point to the following points. (b) shows the orthogonal distances when the RMSE exceeds a threshold, alongside with the first selected vertex V_2 . In (c), the process is repeated starting from V_2 , and a new selected vertex V_3 is shown in (d). similarly, (e), (f) and (g) shown the later selected vertices V_4 and V_5 . (h) shows the final extracted polygon alongside with a new reorder of the selected vertices. 62
- Figure 4.10: Building simplification procedure based on the *SR* showing advantages. (a) showing a building with its reference (yellow polygon). DSM data (b). Building segment (c). MBR (d). Initial and final building outlines are shown in (e) and (f) respectively. 66
- Figure 4.11: Building simplification procedure based on the *SR* showing disadvantages. (a) showing a building with its reference (yellow polygon). DSM

- data (b). Building segment (c). MBR (d). Initial and final building outlines are shown in (e) and (f) respectively..... 66
- Figure 4.12: *Distance-constraint* algorithm for labelling boundary points to corresponding edges. (a) shows a successful implementation while (b) shows its limitation where the black and blue circles demonstrate mislabelling errors. ... 69
- Figure 4.13: *Angle-constraint* algorithm for labelling boundary points to their corresponding edges..... 70
- Figure 4.14: Problem of determination of the closest vertex. The figure shows that the closest vertex to the boundary point P_i is $(V_j - 1)$. This leads to incorrect identification of the previous and next angles..... 71
- Figure 4.15: Boundary point (P) belonging to an extension of edge $(V_1 V_2)$ 72
- Figure 4.16: Gauss-Markov functional model for adjustment of non-rectilinear building polygons. Boundary points p_i which correspond to the adjusted polygon edges (black polygon) are specified (dots in different colours). Coordinates of vertices V_j (black squares) are the unknown parameters u . The vector of observations y comprises the squared distances between boundary points and polygon edges $d^2(i, j)$ (derived observations), cosines of angles α_j and vertices of the polygons V_j 75
- Figure 4.17: Schematic matrix illustration of the GMM equation (4.14) for rectilinear building polygons approximation (modified from (Avbelj, 2015)). Three categories of observation equations were formulated, for squared distances between the boundary points and the building polygon (light blue), for cosines of angles (yellow), and for polygon vertices (green). 76
- Figure 4.18: Covariance matrix Gauss-Markov Model. 77
- Figure 4.19: The orthogonality functional model: Boundary points (dots) coloured differently according to their corresponding edges. Red arrows indicate the direction of normal vectors (not to scale). The blue arrows are the vector distances C computed from the origin in the local building outline coordinate system. The black solid polygon is the final rectilinear polygon after the adjustment. 78
- Figure 4.20: Calculation of RMSE. Red polygon represents a building reference while an extracted building depicted by the green polygon. (a) shows a RMSE calculated from minimum distances measured from each vertex in the reference

to the closest vertex in the extracted and vice versa. (b) considers minimum distances from each vertex to the closest edge or vertex.	84
Figure 5.1: Vaihingen study area including the three Areas 1, 2, and 3 (Cramer, 2010).	87
Figure 5.2: Melville study area.	89
Figure 5.3: The created DSM for Melville dataset. The units of the elevation colour-scale are in metres.	90
Figure 5.4: Area 1 DSM derived from ALS data (a) and reference DTM (b) (Y. Mousa et al., 2017).	93
Figure 5.5: MSD versus NGPs DTM extraction for Area 1(Y. Mousa et al., 2017). (Top) Common input:- DSM derived from image matching is shown in (a), with a 14 cm GSD, as well as the selected ground points (b), and the initial DTM (c). (Middle) MSD algorithm: - The extracted ground mask (d), DTM (e), and nDSM mask (f). (Bottom) NGP algorithm: - extracted ground mask (g), DTM (h), and nDSM mask (i).	95
Figure 5.6: Area 2 DSM derived from LiDAR data (a) and the reference DTM (b) (Y. Mousa et al., 2017).	96
Figure 5.7: DTM extraction for Area1(Y. Mousa et al., 2017). It shows the input DSM derived from image matching techniques (a) with 14 cm GSD, the selected ground points (b), and the initial DTM (c). The second row presents the extracted ground mask (d), DTM (e), and nDSM mask (f) using the MSD algorithm. The extracted ground mask (g), DTM (h), and nDSM mask (i) using the NGPs algorithm are shown in the third row.	97
Figure 5.8: Area 1 elevation difference maps of the LiDAR DTM compared to the MSD (a) and the NGPs (b) (Y. Mousa et al., 2017).	99
Figure 5.9: The DTM evaluation result and analysis for NGPs method. The evaluation result is depicted in (a), where the true positive, false negative, and false positive are showed in yellow, blue, and red respectively. Analysis of some misclassification are presented in (b) and (d), while (c) shows removed cars from terrain. In the reference map, buildings, trees, grass, bare-ground, and cars, are depicted in dark blue, green, cyan, white, and yellow, respectively.	102

- Figure 5.10: Area 2 height differences maps of the created DTMs with LiDAR DTM. MSD (a) and NGPs (b)..... 103
- Figure 5.11: The DTM evaluation result and analysis for NGPs method. The evaluation result is depicted in (a), where the true positive, false negative, and false positive are showed in yellow, blue, and red respectively. Examples of some false positive and false negative are presented in (b) and (c). In the reference, buildings, trees, grass, bare-ground, and cars, are depicted in dark blue, green, cyan, white, and yellow, respectively..... 105
- Figure 5.12: Building detection results in Vaihingen (Y. A. Mousa et al., 2019): (a) Area 1, (b) Area 2, and (c) Area 3. True positives, false negatives, and false positives are depicted in yellow, blue, and red respectively. 108
- Figure 5.13: Building detection in Melville. (a) represents the result for 126 buildings. (b) enlarged section, while (c) and (d) shows one segment building and its nDSM. 110
- Figure 5.14: Procedure for updating buildings map. Orthophoto, old, and new buildings are shown in (a), (b) and (c), while the overlaid old and new map is presented in (d). The DSM is shown in (e). The new and demolished building segments are shown in (f) and (g). The final updated map is shown in (h). 111
- Figure 5.15: Final updated building map for the Melville study area. Unchanged, new, new part, demolished, and demolished-parts buildings are presented in brown, dark blue, light blue, orange, and light green respectively. 113
- Figure 5.16: Example of a complete new building classified incorrectly as a new building-part. The old map, new map, and their overlapping values are shown in the top row. The DSM, Orthophoto, and the result for the updated map from left to right are presented in the bottom row. 114
- Figure 5.17: The sensitivity to a change of the starting point for the DP algorithm (top row) and the proposed method (bottom row) (Y. A. Mousa et al., 2019)..... 116
- Figure 5.18: Comparison of simplification results generated by the DP algorithm and the proposed algorithm (Y. A. Mousa et al., 2019). Triangles denote the starting boundary point. T is the tolerance threshold for DP. V is the number of vertices (\bullet), SD is the calculated standard deviation. Note the RMSE threshold is used in the proposed algorithm..... 118

- Figure 5.19: Sensitivity analysis of the parameters (Y. A. Mousa et al., 2019). (a) sensitivity analysis of *NT* threshold. (b) sensitivity analysis of median filter (*MF*), RMSE threshold (*RT*), angle detector (*AD*), and similarity ratio (*SR*)..... 120
- Figure 5.20: Building regularisation result in Vaihingen dataset (Y. A. Mousa et al., 2019). Areas 1, 2, and 3 from left to right. Building colours are randomly assigned for better visualisation..... 123
- Figure 5.21: Building regularisation result in the Melville dataset. Building colours are randomly assigned for better visualisation..... 124
- Figure 5.22: The calculated RMSE in pixels of the extracted polygons before and after the adjustment – Melville dataset..... 125
- Figure 5.23: Improvement of the RMSE after the adjustment – Melville dataset. .. 125
- Figure 5.24: Assessment of rectilinear building polygons processed by GMM – Melville dataset (a) shows the RMSE of the approximated and adjusted polygons. (b) presents the percentage improvement of the RMSE planimetric accuracy. 127
- Figure 5.25: Examples show improving planimetric accuracy of rectilinear buildings processed by GMM. First column (a), (d) and (g) depict the approximated polygons (green) and the reference (red polygons) before adjustment. Second column (b), (e) and (h) represent the final extracted polygons after adjustment (green) along with references. The third column present nDSM and binary image. 128
- Figure 5.26: Examples show the declining planimetric accuracy of rectilinear buildings processed by GMM. First column (a) and (c) depict the approximated polygons (green) and the reference (red polygons) before adjustment. Second column (b) and (d) represent the final extracted polygons after adjustment (green) along with references (red). (a1) and (d1) are images taken from Google Maps while (c1) is the input building segment as extracted from the nDSM..... 131
- Figure 5.27: Assessment of non-rectilinear building polygons processed by GMM presenting the percentage of the improvement of the RMSE planimetric accuracy. 132
- Figure 5.28: Examples show improving planimetric accuracy of non-rectilinear buildings processed by GMM. First column (a), (c) and (f) depict the approximated polygons (green) and the reference (red polygons) before

- adjustment. (b), (d) and (f) represent the final extracted polygons after adjustment (green) along with references. Images taken from Google Maps are also presented. 133
- Figure 5.29: Examples show declining planimetric accuracy of two non-rectilinear buildings processed by GMM. Image taken from Google Map (a1), nDSMs, and building segment are also shown. 135
- Figure 5.30: Assessment of non-rectilinear building polygons processed using the GHM adjustment. (a) shows the RMSE of the approximated and the adjusted polygons. (b) presents the percentage of the improvement of the RMSE planimetric accuracy. 136
- Figure 5.31: Examples show improving planimetric accuracy of two non-rectilinear buildings processed by GHM. (a) and (c) represent the approximated polygons (green) and the reference polygon (red) before adjustment. (b) and (d) represent the final extracted polygons after adjustment (green) along with references polygon (red). 137
- Figure 5.32: Examples with declining planimetric accuracy of non-rectilinear buildings processed using the GHM adjustment. First column (a) and (c) show the approximated polygons (green) and the reference polygons (red) before adjustment. Second column (b) and (d) represent the final extracted polygons after adjustment (green) along with references (red). 138
- Figure 5.33: Percentage of the improvement of the planimetric accuracy of rectilinear building polygons processed using the GHM adjustment. 139
- Figure 5.34: Rectilinear buildings processed using the GHM adjustment. The first example showing a building presented in (a), (b), and (c) showing the building before the adjustment, after the adjustment and overlaid with the orthophoto respectively. The reference is always shown as a red polygon. The second example is a building presented in (f) and (g) before and after the adjustment, respectively. The reference is shown in red. The building polygons are overlaid with images taken from Google Maps (f1) and the nDSM (g1). 140
- Figure 5.35: Three rectilinear buildings with improved planimetric accuracy processed using the GHM adjustment. The extracted building segments (a), (c),(e) before

adjustment; and approximated polygons (b), (d),(f) The reference dataset is shown in red.	142
---	-----

List of tables

Table 2.1: Names and centre wavelength of spectral band (MicaSense, 2017).....	11
Table 5.1: Parameters' values (MSD approach).	92
Table 5.2: Parameter values (NGPs approach).	92
Table 5.3: Area 1 statistics of Mean Square Error (MSE) and standard deviation (McAndrew) of the height differences of MSD and NGPs compared with the LiDAR DTM as well as the time required to execute the algorithms.....	100
Table 5.4: Area 2 statistics of Mean Square Error (MSE) and standard deviation (McAndrew) of height differences between MSD and NGPs compared with the LiDAR DTM and time required to execute the algorithms.	104
Table 5.5: Area-based and object-based evaluations of building outline extraction output. C_m = completeness, C_r = correctness, Q_l = quality.	106
Table 5.6: Parameters setup for building regularisation.	121
Table 5.7: Evaluation results of building reconstruction: area-based completeness (C_m), correctness (C_r), and quality (Q_l), RMSE of the extracted vertices and RMSE of centroids of building objects as well as the processing time in seconds.	122
Table 5.8: Model assumption evaluation – Melville dataset	126

Acronyms

Acronym	Meaning
2D	Two-Dimensional
3D	Three-Dimensional
ALS	Airborne Laser Scanner
BP	Building Polygon
DSM	Digital Surface Model
DTM	Digital Terrain Model
DP	Douglas-Peucker
GHM	Gauss-Helmert Model
GMM	Gauss-Markov Model
GNSS	Global Navigation Satellite System
GSD	Ground Sampling Distance
IMU	Inertial Measurement Unit
LiDAR	Light Detection and Ranging
MBR	Minimum Bounding Rectangle
MSD	Multi-directional and Slope Dependent
nDSM	Normalised Digital Surface Model
NDVI	Normalised Difference Vegetation Index
NGPs	Network of Ground Points
NIR	Near Infrared
PoLiS	Polygons and Line Segments
RANSAC	Random Sampling Consensus
RGB	Red-Green-Blue
RMBR	Recursive Minimum Bounding Rectangle
RMSE	Root Mean Square Error
RS	Remote Sensing
RT	RMSE Threshold
SGM	Semi-Global Matching

SR	Similarity Ratio
V2M	Vertex to Model
V2V	Vertex to Vertex

1 CHAPTER ONE INTRODUCTION

1.1 Building detection and boundary regularisation

Automating the process of building detection and boundary regularisation has been a topic of interest in Photogrammetry and Remote Sensing for over three decades. This process is required for several applications such as map production (Höhle, 2017) and updating (Awrangjeb, 2015), 3D building modelling (Vosselman & Maas, 2010), and real-estate databases (Awrangjeb, 2015). Due to increasing urbanisation globally, this has increased the demand for generating new and updating existing building footprints and databases within a reasonable time, cost, and accuracy (Bulatov et al., 2014). Therefore, automating this process is essential in order to meet these demands and requirements.

The initial step for creating or updating building outline databases is the detection of building objects, followed by extracting the outline of the detected building. Numerous methods have been developed in the last decades for extracting building objects. These methods can be grouped in terms of the data sources, and whether they are being extracted from a single data source, or multiple data sources. For instance, there are approaches using single imagery data only (Huang & Zhang, 2011; Qin & Fang, 2014), Light Detection and Ranging (LiDAR) data only (Awrangjeb, Lu, & Fraser, 2014), or combining multiple datasets (Awrangjeb, Ravanbakhsh, & Fraser, 2010; Rottensteiner, Trinder, Clode, & Kubik, 2007). However, elevation data, represented by digital surface models (DSMs) for example, is essential input in addition to imagery data in all cases.

With the rapid development of sensor technology, very high-resolution imagery with high overlap has become accessible without additional cost. Alongside the development of dense image matching algorithms (e.g. semi-global matching (SGM) (Hirschmuller, 2008)), there is the possibility for accurate DSM generation. Occlusions (e.g. through the view angle in urban corridors or through trees along urban

streets), and shadow effects are the main limitations of imagery-based DSMs (Kwak, 2013). LiDAR systems can overcome those limitations, and are therefore an important imaging source for acquisition of accurate DSMs. The disadvantages of LiDAR, however, are the costs for the data acquisition as well as difficulties for capturing semantic object information (Kwak, 2013) and data with high spatial density.

Nevertheless, the majority of existing methods tend to combine multiple geospatial data sources. This usually leads to increased accuracy with respect to the building object identification and modelling (Habib, Ghanma, & Mitishita, 2004). This is because the limitations of certain datasets can be overcome by another, which leads to better objects classification and modelling results.

1.2 Problem statement

Building detection in terms of existing procedures starts with the extraction of a digital terrain model (DTM), which is the digital representation of the bare ground surface without man-made objects, from one or more DSMs (Bulatov et al., 2014; Mongus, Lukač, & Žalik, 2014). Algorithms for DTM extraction are based either on “raw” point cloud DSMs or on rasterised DSMs. In both cases, the following physical characteristics of terrain surface have been considered to identify points belonging to the terrain: (i) lowest elevation in a local area (if there are no outliers); (ii) slope angle; (iii) elevation difference; and (iv) surface homogeneity (Meng, Currit, & Zhao, 2010). A central focus of this thesis is to discuss and analyse the strengths and drawbacks of the implemented parameters, which effect the estimation of the aforementioned physical characteristics. Slope in particular is the most sensitive parameter in the DTM extraction process (Zhang & Whitman, 2005). Furthermore, determining a slope threshold in terms of terrain information in the analysed scene is, somewhat, subjective (Zhang et al., 2003). Thus, further investigation and research to develop more efficient solution, especially considering increasingly dense data, is required.

After DTM extraction the normalised DSM (nDSM), representing buildings, trees and other non-ground objects, can be created by subtracting the DTM from the DSM. It can be challenging delineating trees from the nDSM, especially those located in shadow regions, even with the availability of Near-Infrared images (Gerke and Xiao,

2014). To overcome this limitation, the planarity measure (West et al., 2004) within a robust data fusion procedure for differentiation of buildings from trees is possible. An alternative method of extracting buildings to the process described above is via image classification. More recently, significant progress has been made utilising machine learning approaches such as convolutional neural networks (CNNs) (Long, Shelhamer, & Darrell, 2015). Its advantage is in its effectiveness in image classification with pixel-level accuracy. The main impediment to greater utilisation, however, lies in the expensive training effort required in order to be able to create the classifier. Further, results are presented in raster form, and usually need to be converted to a vector format for end-use, such as the updating of maps and databases, or the modelling of building. The linked problems of improving the classification training and vectorisation of raster data by this approach have received little research focus (Avbelj, 2015) and are another important focus of this thesis.

The process of generating this vector data is also called the regularisation of building boundaries and is more precisely defined as the procedure for generating approximate building polygons and their refinement. Existing building outline regularisation algorithms can be grouped into three categories: (i) model-driven (Brédif, Tournaire, Vallet, & Champion, 2013; Sohn, Jwa, Jung, & Kim, 2012) ; (ii) data-driven (Pohl, Meidow, & Bulatov, 2017) : or (iii) a combination of model- and data-driven (He, Zhang, & Fraser, 2014).

Model-driven approaches rely on several preselected parametric building models to be fitted with a given boundary-point dataset; they are therefore more robust against noise (Kwak, 2013). Such approaches are based on the orthogonality characteristics of building outlines (right-angled corners), which is true for most existing buildings but can be a challenge for complex buildings. Therefore, non-rectangular buildings can be incorrectly simplified or be represented by overly complex shapes (Avbelj, 2015).

In contrast to the model-driven approaches, data-driven approaches do not require prior knowledge of the building shape; they therefore offer greater flexibility to model any building shape. Nevertheless, data-driven approaches still face difficulties in the

case of missing significant data (Sohn et al., 2012) and it is often difficult to formulate constraints imposing regularity (Kwak & Habib, 2014). Many existing data-driven approaches have utilised an initial solution based on the data-driven (Douglas-Peucker (DP)) algorithm (He et al., 2014; Jwa, Sohn, Tao, & Cho, 2008; Maas & Vosselman, 1999; Sohn et al., 2012; Wang, Lodha, & Helmbold, 2006) because it is easy to implement and is able to maintain the original shape (Song & Miao, 2016). However, the efficiency of the DP algorithm decreases with increasing irregularity of the building boundary points. This is because the algorithm is inherently sensitive to both noise and the position of the starting point. In addition, building characteristics such as angle detection and area preservation are not considered in the processing. Therefore, a more robust data-driven approach is preferable.

Following nDSM creation the building polygons must be derived. The approximate building polygons generated from model-driven approaches usually have regular shapes. Best-fitting of polygon edges to the input boundary points is required to enhance the planimetric accuracy of building outlines (Avbelj, 2015). Additionally, the dominant building direction of the approximate building polygon is not necessarily accurate (He et al., 2014). The approximate building polygons generated from a data-driven approach mostly have irregular shapes. Thus, further improvement in terms of regularity and best-fitting is required. For instance, to refine the building polygons, the orthogonal distances between building edges should be measured and minimised in a least squares adjustment. Perpendicularity constraints of adjacent building edges must be considered in the adjustment if a building is rectilinear which is mostly the case. The challenge is to adjust all parameters simultaneously (Avbelj, 2015).

1.3 Objective of the thesis

This thesis aims to develop a fully automatic workflow for regularisation of building footprints, overcoming the problems highlighted previously. This work follows a typical process for 2D building mapping and consists of three crucial steps: extraction of the DTM from the DSM, the building segment detection, and boundary regularisation. Accordingly, the objectives of this thesis are:

- (i) Review of existing DTM extraction algorithms to identify their shortcomings; and based on the findings,
- (ii) Develop a new DTM extraction algorithm for the processing of high-resolution DSMs;
- (iii) Design a data fusion approach to enhance the extraction of buildings. (This includes the generation and processing of a nDSM, a planarity map, and a Normalized Difference Vegetation Index (NDVI) map. These results will be fused in a pixel-wise classification procedure.); and
- (iv) Combine the robustness of the model-driven approaches with the flexibility of the data-driven approaches in order to build a comprehensive strategy to deal with building regularisation problem more effectively. This includes:
 - 1. Developing a new data-driven approach to generate approximate polygons for rectilinear and non-rectilinear buildings.
 - 2. Solving boundary regularisation of rectilinear buildings by implementation of an orthogonality constraint and best fitting of boundary points w.r.t. their corresponding edges simultaneously in the adjustment process.
 - 3. proposing a new approach for automatic updating of the assignment/label of the boundary points to their corresponding edges during the adjustment.
 - 4. Introducing a new evaluation method for assessing corresponding polygons (i.e. extracted and its reference polygons) utilising the Root Mean Square Error (RMSE) as a standard topographic measure.

1.4 Research methodology

To achieve the objective of this research, a detailed literature review has been conducted in terms of DTM extraction, building detection and regularisation in order to highlight the shortcomings/limitations in existing methods. Then, a new methodology is proposed to overcome these identified limitations. Firstly, LiDAR point cloud data is cleaned to remove outliers, and then rasterised. Then a new DTM extraction algorithm is proposed to cope with shortcomings of existing methods and

create a high quality nDSM. Additionally, nDSM, Planarity map and NDVI are fused to define the building segment. Furthermore, approximate building polygons are created for these segments. This step includes a novel data-driven method for creating simplified polygons for rectilinear and non-rectilinear buildings. Finally, GMM and GHM adjustments are implemented to create the final building outlines.

1.5 Significance of the research

The automation of building detection and boundary regularisation procedure has several stages. Most current approaches have one or more limitations in the workflow. Significant limitations include: limitation of DTM extraction from very high resolution DSM; eliminating trees located in shadow regions; generation of approximate building polygons without considering building characteristics; estimating the dominant building orientation; and challenges in solving regularisation of rectilinear building outlines.

The significance of this study is to provide a forthright workflow that overcomes these limitations for large scale areas, and at the same time achieving sub-metre accuracy level of building outlines. The workflow introduces a new and robust evaluation method to measure the absolute accuracy between a polygon and its reference utilising the RMSE as a standard assessment tool.

Significant improvement has been achieved in term of objects classification. This work proposes and demonstrates a workflow that enables a faster but at the same time still precise method to extract building outlines from high resolution DSM data. The results contribute to the science of creating and updating vector datasets of building outlines and for the creation and updating of topographic maps in urban areas.

1.6 Thesis organisation

The thesis is divided into six chapters as follows:

Chapter 2 explains the theoretical background and starts with a brief description of LiDAR systems, including the main mechanisms and properties. Then, the basics of

imagery data and their derived and relevant information are outlined. Data representation and processing including transformation of 3D point clouds to gridded form, planarity generation, and basic mathematical morphology operations are introduced. Physical characteristics of the ground surfaces are then discussed. The mathematical models for building polygons regularisation will be presented at the end of this chapter.

Chapter 3 presents related works in terms of the data sources and processing strategies used, together with detailed review of deficiencies in the existing strategies.

Chapter 4 describes the proposed workflow methodology in detail. The chapter starts with the pre-processing of laser scanned points, particularly identifying and removing of outliers as well as the transformation of those laser scanned points into a regular gridded DSM. The proposed DTM extraction algorithm is introduced followed by the proposed data fusion procedure for extracting building segment. A novel data-driven algorithm for creating approximate building polygons is then described. Finally, mathematical models for regularisation of the approximated building polygons are implemented using the concept of least squares adjustment. This step includes a newly developed algorithm for automatic updating the labelling of boundary points to their corresponding polygon edges in the iterated least squares.

Chapter 5 presents the trial results utilising the proposed methodology for selected test areas as well as an accuracy evaluation and discussion.

Chapter 6 provides a summary of the thesis conclusions achievements of the proposed methods, and recommendations for future work.

2 Chapter two Background

2.1 Overview

This chapter introduces the theoretical background and characteristics of point clouds created from LiDAR and Remote Sensing (RS) relevant for building detection and outline regularisation algorithms. This introduction is required in order to be able to discuss the shortcomings of the existing building detection algorithms introduced in the next chapter.

2.2 Point Cloud creation

2.2.1 LiDAR

LiDAR scanners are active sensors emitting pulses of light that illuminate a spatial swath of interest, and discretely samples the range from the source to points in the field of view. The distance from objects in the field of view to the scanner is calculated based on the time that the laser light needs to return to the receiver. While the laser light is used to determine the distance of the Airborne Laser Scanner (Alsadik & Remondino) system to the object, the position of the scanner has to be known in order to derive three dimensional coordinates of the object in space. An ALS system consists therefore of the LiDAR system, Inertial Measurement Unit [IMU] and a Global Navigation Satellite System (GNSS) for positioning. Figure 2.1 illustrates the basic components of an ALS. Using the known position and rotation of the receiver together with the calculated time and the speed of light, the position of captured points is calculated. Positioning errors of the ALS system can therefore lead to inaccurate point clouds. Those effects are usually overcome in the post-processing stage, using methods to correct the errors such as strip adjustment (Abed, 2012). One common limitation inherent to LiDAR datasets are data “holes” which are usually caused by shiny planar surfaces which reflect the light away from the scanner, or light-absorbing surfaces which allows no light to be reflected.

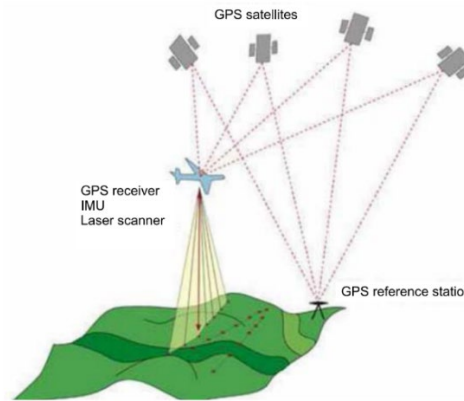


Figure 2.1: Basic components of an ALS (Arefi, 2009).

Two other types of errors can be found in the LiDAR measurements which are horizontal and vertical errors. The horizontal errors are mainly caused by the angular uncertainty of the IMU, which linearly depends on the flying height, and reflection of light from sloped terrains (Hodgson & Bresnahan, 2004). Though the measured distance from a sensor to an inclined terrain is accurate, the planar error of such measured response leads to an obvious systematic error in the measured distance (Hodgson & Bresnahan, 2004). This error can lead to the same effects as relief displacement in optical images and it is demonstrated in boundaries of elevated objects (Avbelj, 2015). The vertical error is significantly lower than the horizontal error and it is mainly caused by the LiDAR sensor platform properties, e.g. determination of the return location in the pulse length, and the efficiency of GNSS/IMU (Hodgson & Bresnahan, 2004).

The acquired 3D points can have inconsistent outlier observations. These outliers are points with unreliable elevations compared with the whole data or in particular, with their local neighbourhood (Arefi, 2009). Such outliers are usually removed based on the normal distribution of the data. Thus, the acquired 3D points may have irregularly-spaced distribution after outlier point removal.

One of the important characteristics of LiDAR systems is the ability to record multiple echo returns, i.e. when the laser returns to the receiver, not just the first returning echo but also all following returning echoes, or the full returned waveform can be recorded

(Abed, 2012). In areas of vegetation, due to the spot size of the laser and the ability to register more than one returning echo to the receiver, laser pulses that are reflected on the top of the tree crown are registered as first echo returns while the laser pulses that are reflected off the ground beneath the tree are registered as last returning echo. Laser pulses have an advantage related to the ability to penetrate a tree canopy, as shown in Figure 2.2. This characteristic can be used to identify vegetation (but incompletely) and, therefore, to remove those above-ground objects from the point cloud. Remaining elevated objects in the point cloud are man-made structures such as buildings next to ground points. However, it has been shown that vegetation cannot be well-distinguished from man-made objects (i.e. buildings) using ALS data (Yoon, Shin, & Lee, 2008).

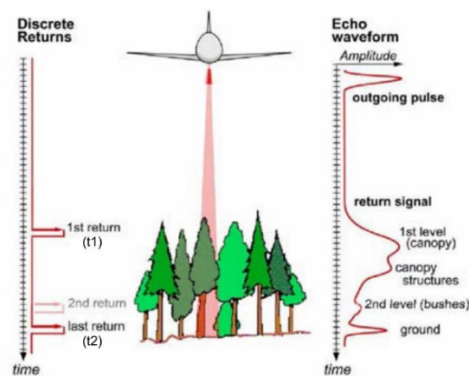


Figure 2.2: Multiple reflection laser observation (Arefi, 2009).

2.2.2 Imagery

Images are 2D data from a passive sensor and can be captured from satellites, airborne platforms, Unmanned Aerial Vehicle (UAV), or terrestrial (e.g. close range photogrammetry). If two or more overlapping images are taken from different positions, stereo-photogrammetric methods can be utilised to generate 3D measurements. This 3D information is necessary in order to create ortho-rectified images, which is one of main outcomes of the photogrammetric processing of images. Images captured from different remote sensing platforms will vary in terms of their geometric accuracy, spatial resolution (Ground Sample Distance, GSD) as well as types and number of bands captured. A higher resolution image usually allows denser and more detailed point clouds to be derived compared to low resolution imagery

(Hastedt & Luhmann, 2015). Larger image size are able to create higher accuracy for the projection and interior orientation parameters (Conte, Girelli, & Mandanici, 2018). Furthermore, the accuracy of estimating interior and exterior parameters of airborne cameras may generally be higher compared to UAV cameras due to lack of stability and long-term validity of the interior orientation parameters of the cameras used on UAV platforms (Hastedt & Luhmann, 2015).

The wavelengths for bands, as well as number of bands can vary depending on the sensor. Table 2.1 illustrates names and centre wavelength of the most common spectral bands used. Blue, Green, and Red band present the visible light. Red edges (RE) and Near-infrared (NIR) bands are suitable to detect vegetation through computing of the Normalized Difference Vegetation Index (NDVI). The NDVI is calculated as follows.

$$\text{NDVI} = \frac{(\text{NIR} - \text{Red})}{(\text{NIR} + \text{Red})} \quad (2.1)$$

Table 2.1: Names and centre wavelength of spectral band (MicaSense, 2017).

Band	Spectral Bands	Centre Wavelength (nm)
B1	Blue	475
B2	Green	560
B3	Red	668
B4	Red edge	717
B5	Near-infrared	840

2.2.2.1 Image-based derived 3D point clouds

Photogrammetric stereo-plotting and dense image matching are the most common methods of acquiring 3D point clouds from overlapping images. The interior and exterior orientation parameters must be estimated first. Several image matching algorithms have been developed in the last decades to find the correspondence points in the overlapped images with sub-pixel accuracy (Leberl et al., 2010). Image matching methods can be generally grouped into two groups: feature-based or area-

based (Hong & Zhang, 2007). The feature-based image matching tends to automate the procedure of the relative information, while the area-based methods often need known interior and exterior camera calibration parameters.

For wide baseline application, the most efficient feature-based matching algorithm, known as the Scale-Invariant Feature Transform (SIFT) (Lowe, 1999), is used due to its robustness against occlusion, variable illumination intensity, and scaling. This algorithm does not need segmentation (Bastanlar, Temizel, & Yardimci, 2010; Mikolajczyk & Schmid, 2005). From available camera positions and image correspondences, points in space can be triangulated (Hartley & Zisserman, 2000). However, even in the case for which the interior and exterior orientation parameters of camera are unknown, they can be computed within a system of equations followed by bundle adjustment (Hartley & Zisserman, 2000). However, in the absence of orientation parameters the result of this step is a sparse point cloud. However it is necessary to obtain a dense point cloud for the task of object detection (Bulatov, Wernerus, & Heipke, 2011), and ideally it should be every pixel in every image. The distance from the point to the camera principal plane is called depth and the depth map is the set of depth for all pixels.

Computation of high quality depth maps is challenging mainly due to the occlusions, reflections, regions of homogeneous texture, repetitive patterns, and slanted surfaces (Bulatov et al., 2011). The technique Semi-Global-Matching (Hirschmuller, 2008) allows for a fast and accurate computation of depth or disparity. This method consists of two steps: firstly, a cost function can be computed by means of the Sum of Absolute Difference (Alsadik & Remondino), Sum of Squared Difference (SSD) or a robust matching cost against the radiometric distortions such as Mutual Information (MI) or Normalized Cross Correlation (NCC) (Bethmann & Luhmann, 2014). The second step required is the cost aggregation or Non-local optimization. The values of the cost function are estimated for each individual pixel and depth value via aggregated path costs. The paths run radially from the image border to the center. The individual cost for each running path is accumulated and the depth with minimum cost is selected. A minimum of eight paths are needed to provide satisfactory depth map (Hirschmuller,

2008). Figure (1) shows an example of estimating disparity value from 16 aggregated paths from all direction towards the pixel under evaluation.

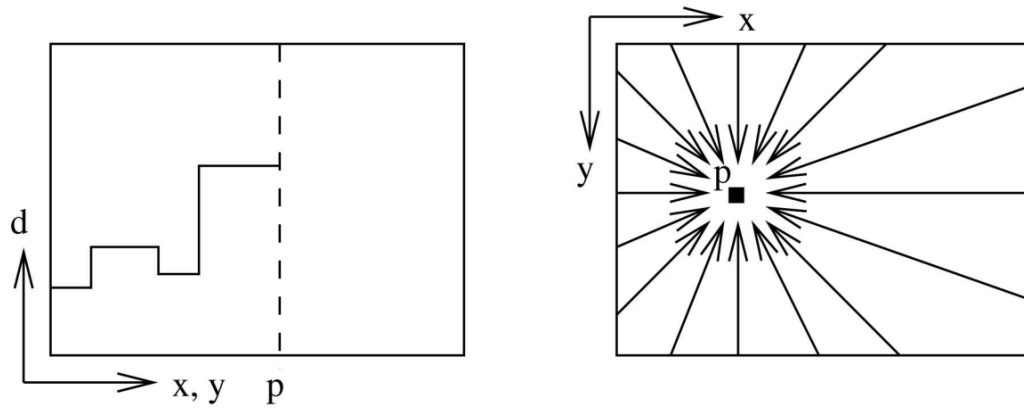


Figure 1: cost aggregation in disparity space. Left to right: Minimum cost path and example of 16 paths from all direction (Hirschmuller, 2008)

2.2.2.2 Orthophoto images

Optical images suffer from relief displacement problems. The relief displacement is defined as a change in an object's location in an image caused by the object's altitude above/below the terrain and viewing direction of the sensor (Abed, 2012). The effect of this displacement increased with increasing angle to nadir, and increasing the object's height (L. Chen, Zhao, Han, & Li, 2012). A displacement of pixels, e.g. belonging to the top of a high building, can be also be visible in the ortho-images due to height errors in the underlying DSM used in the ortho-image production (see Figure 2.4 a). Hence, those points may not represent the actual projected position on the map geometry (see Figure 2.3(a)). The production of ortho-images requires knowing the camera orientation parameters, and having a digital surface model (DSM). The DSM should have the same resolution as the image in hilly terrains (Passini & Jacobsen, 2004). The orthogonal projection for each pixel in the image is computed using the collinearity equations.

The use of ortho-images is demonstrated by the fusion with other data sources such as LiDAR, especially for building detection purposes however alignment of the

orthophoto with other data sources may still not be satisfactory (L. Chen et al., 2012). Consequently, false negative error due to the misalignment can occur in the building extraction result (Sohn & Dowman, 2007). Figure 2.3(a) and (b) illustrates an image before and after rectification, respectively. The rectified image (Figure 2.3(b)) might be well-aligned but some black holes caused by occlusion can be seen.

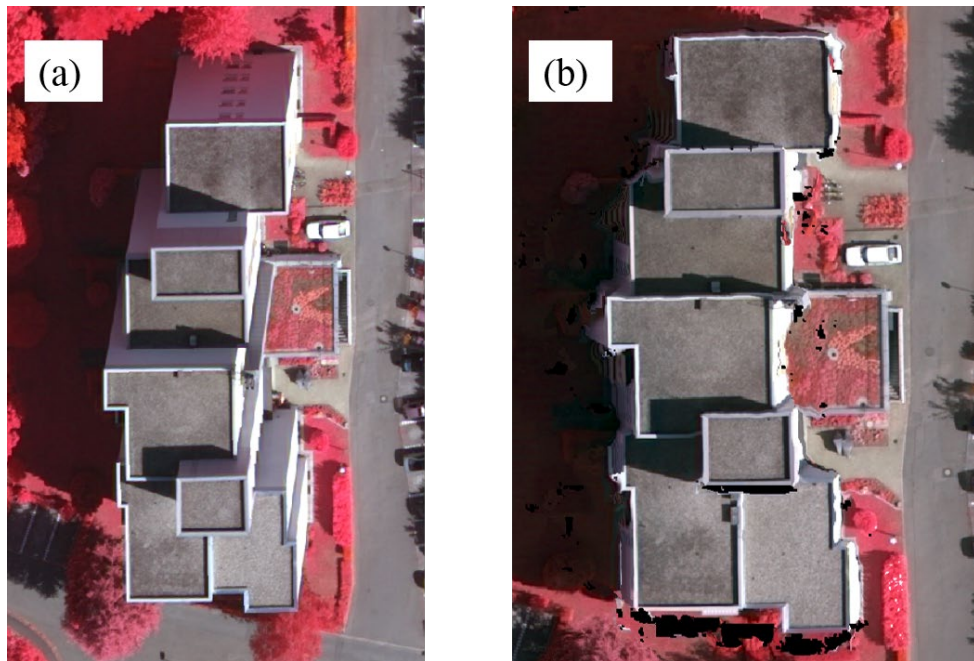


Figure 2.3: Generation of orthophoto images. (a) raw image. (b) rectified image provided by the German Society for Photogrammetry, Remote Sensing and Geoinformation (DGPF) (Cramer, 2010).

2.3 Discussion of Lidar versus image-based point clouds

The quality of the generated image-based point clouds is highly dependent on the success of the image matching. For example, in shadow regions or occluded parts of buildings, it is very hard or impossible to find the correct position of corresponding pixels, leading to matching errors or gaps. Hence, gross errors can occur in the calculated point clouds due to the incorrect trigonometry (Bergsjö, 2016). In contrast, ALS systems outperforms photogrammetric based methods by overcoming the shadow effect, due to being non-passive and independent of solar illuminance (Abed, 2012) and lowering the influence of occlusions. However, laser-scanned points usually contain outliers. Two categories of outliers may exist, namely wide/low outliers and

broad/high outliers. The wide outliers are usually caused by refraction of light from elements in an atmosphere, power lines, or birds, while broad outliers are mainly caused by multiple reflections (Arefi, 2009). The type of the target material plays an important role on light reflection, e.g. water bodies. Consequently, the reflected laser light can be too weak or never be observed by the receiver due to specular reflection from the target material, leading to holes (missing data). In image-matching outliers can also occur and are mainly caused by occlusion. This appears when a point is visible in one image only. The outliers can be either global or local. Global outliers are mainly caused by mismatching of corresponding points or insufficient camera calibration, while local outliers are persuaded by random deviations and uncertainties in the camera position and depth map approximation process (e.g., depth quantisation) (Stucker, Richard, Wegner, & Schindler, 2018).

In a comparison study presented by (Leberl et al., 2010), the authors argue that innovative photogrammetric methods provide point clouds with comparable accuracy but higher density than ALS system. Regarding building outlines, the accuracy is highly depend on the density of the point cloud (Sampath & Shan, 2007). In terms of automation, LiDAR point clouds can be processed straight-forwardly while user interaction might be needed in stereo photogrammetry (Kwak, 2013). For more details, a comparison study between LiDAR and imagery point clouds was given in Leberl et al. (2010).

2.4 Data rasterisation and processing

2.4.1 Transformation of 3D point clouds to gridded DSM

In order to apply standard image processing methods such as filters for the detection of buildings, it is required to turn the 3D point cloud into a raster image (also called gridded DSM). The raw 3D laser scanned points are randomly distributed and need to be organised in such a way that information can be extracted from them (e.g. building detection). Therefore, interpolating them into gridded data is the preferable option (Ma, 2004). The process of transforming or interpolation of 3D point clouds into a gridded DSM or raster image is called data rasterisation. The output is an image where

the value of each pixel corresponds to the elevation height of this grid. To convert point clouds to a raster image, a 2D transformation is applied from the object plane coordinate system (X, Y) to the image coordinated system (r, c) . Here, X corresponds to c in the horizontal direction, while Y corresponds to r in vertical direction (see Figure 2.4). The top left pixel (r, c) is the origin $(0, 0)$ of the image coordinate system. If laser scanned points were captured diagonally, an internal rotation could be used to conserve space.

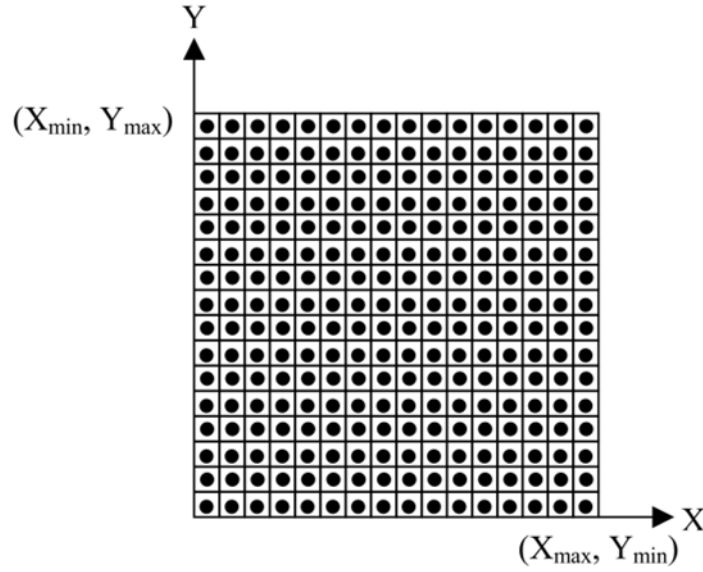


Figure 2.4: Transformation from point clouds to gridded-DSM (Ma, 2004)

The transformation from the 3D coordinate system (X, Y, Z) to the 2D image coordinate system (r, c) can be performed as follow:

$$\text{image}(r, c) = \text{integer} \left(\frac{Y_{\max} - Y}{GSD}, \frac{X - X_{\min}}{GSD} \right) \quad (2.2)$$

Each pixel (r, c) in the image has the Z value that corresponds to the X and Y that fall into the calculated position as in the equation above. GSD is the desirable ground sample distance for the gridded image. The minimum GSD can be estimated from the point cloud average density as $1/\sqrt{n}$ (Ma, 2004), where n is the average number of laser scanned points located in one metre square. When more than one point fall into a grid cell, usually the minimum value is considered assuming that the minimum value corresponds to the terrain (Ma, 2004). If there are no points located in a grid cell, interpolation methods such as linear interpolation can be applied. The rasterisation

process may introduce smoothing error depending on the selected GSD, but the effect is usually low (Hodgson & Bresnahan, 2004).

2.4.2 Planarity

For further processing, such as when analysing the height differences of a point and its neighbours in a gridded DSM using a pre-defined window size, several local geometric measures can be obtained. One example is the planarity measure, which, essentially assesses how the neighbourhood of a point can be approximated by a plane (West et al., 2004). To describe the local 3D construction nearby a 3D point $X = X_o$, the so-called structure tensor C is obtained from the set of neighbouring 3D points X_i with

$i = 1, \dots, n$ by considering the respective 3D covariance matrix as follows (Weinmann, Urban, Hinz, Jutzi, & Mallet, 2015):

$$C = \frac{1}{n+1} \sum_{i=0}^n (X_i - \bar{X}) (X_i - \bar{X})^T \quad (2.3)$$

$$\bar{X} = \frac{1}{n+1} \sum_{i=0}^n X_i \quad (2.4)$$

To compute the structure tensor, it is preferable to interpret the slightly smoothed DSM as a point cloud, and based on the cylinder-like extraction of neighbours proposed by Gross and Thoennessen (2006). The smoothing can be done using a median or Gaussian filter. Different measures can be obtained from the eigenvalues (λ_1, λ_2 , and λ_3) of the structure tensor C , where $\lambda_1, \lambda_2, \lambda_3 \in R$ and $\lambda_1 \geq \lambda_2 \geq \lambda_3 \geq 0$. Accordingly, the planarity measure P_λ can be calculated as follows:

$$P_\lambda = \frac{\lambda_2 - \lambda_3}{\lambda_1} \quad (2.5)$$

Then, the planarity map can be visualised as a float image, where each pixel value represents a planarity measure. An example is given in Figure 2.5 where the pixel values of the planarity measures are scaled from 0 to 1 and colour coded. It is clearly visible that the planarity measures of the terrain are very high (close to 1) followed by

the planarity measure of the roofs with a measure of approximately 0.8. In contrast the planarity measure of the trees and the building edges are very low with a value less than 0.2. Hence, a separation of planar man-made objects and the terrain compared to vegetation is possible.

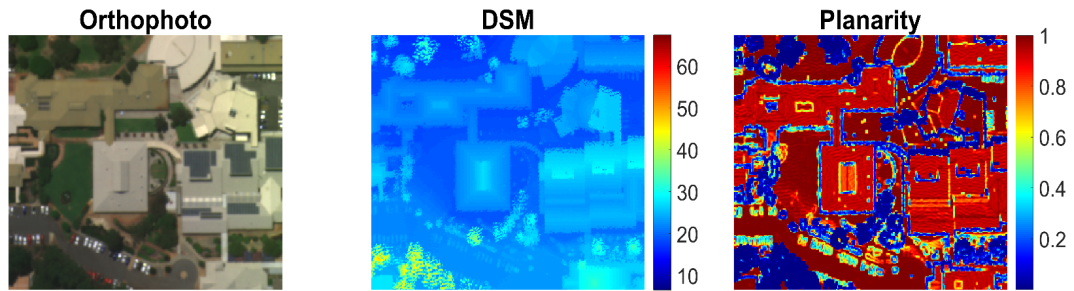


Figure 2.5: Planarity measure. Orthophoto, DSM, and the planarity map from left to right.

2.4.3 Mathematical Morphology

When elevation data are represented as a greyscale image, the image processing tools, including morphologic operators using a structural element (SE) can be utilised. Such morphological operators are erosion and dilation (Haralick, Sternberg, & Zhuang, 1987), which allow the extraction of terrain points, this is, points belonging to the digital terrain model.

Searching for neighbouring pixels can be performed using filtering procedures within a predefined size SE . The SE can be described with respect to different shapes (e.g. disk or square) and with a pre-defined size. Dilation ($\delta_{SE}(X)$) and erosion ($\varepsilon_{SE}(X)$) of X (where X is defined as set of pixels having values of one) with a structure element SE are the principle morphological operators. Dilation means increasing an object area in all directions within a pre-defined size and shape of the SE . In other words, it is an extension of bright pixels (pixels having values equal to one in a binary image). In contrast, erosion means decreasing or shrinking an object area in an image. In a binary image, the mathematical description of dilation and erosion (Arefi, 2009) are:

$$\delta_{SE}(X) = \{x | SE_x \cap X \neq \emptyset\} \quad (2.6)$$

$$\varepsilon_{SE}(X) = \{x | SE_x \subseteq X\} \quad (2.7)$$

Equation (2.6) describes the dilation where the position of pixel x such that SE hits X when its centre overlaps with x . In contrast, equation (2.7) represents the erosion, where the position of pixels x such that SE is included in X when its centre is located at x . Figure 2.6 illustrates an example of dilation and erosion using a binary image. Very small objects could disappear completely if a large SE is applied. Accordingly, many other operators can be modified, such as opening and closing in binary and greyscale images (Arefi, 2009). Nevertheless, characteristics of an object including area, size, centroid, minor and major axis can be derived by analysing the connected component of the output image.

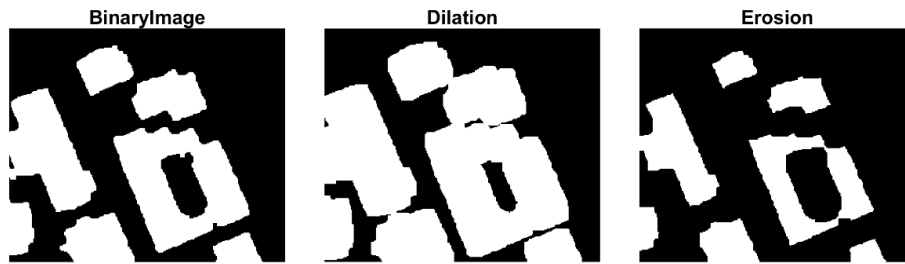


Figure 2.6: Morphological operators. From left to right: binary image shows building segments, example of dilated image, and example of eroded image. Buildings are represented as white pixels; terrain is shown with black pixels.

2.5 Ground Characteristics for DTM extraction

Digital Terrain Models only include bare-ground regions without any man-made objects such buildings. With this definition, hills or cliffs should be part of the DTM. Extracting the DTM is the first step in the building detection procedure. In order to correctly identify pixels belonging to the terrain the physical characteristics of ground surface should be understood. These characteristics can be summarised as follows (Meng et al., 2010):

- (i) Minimum elevation: ground pixels usually have lower elevation than non-ground pixels in a local search. Several DTM extraction algorithms are based on this characteristic.
- (ii) Slope measure: the slope among ground pixels is commonly lower than the slope between ground and non-ground pixels (Zhang & Whitman, 2005).
- (iii) Height difference: adjacent ground pixels generally have lower height differences than the ground and non-ground pixels. Hence, pixels having higher height differences than a pre-defined threshold are commonly classified as non-ground pixels, such as buildings or trees.
- (iv) Ground surface homogeneity could be also a possible option to distinguish ground from non-ground pixels, because the former are assumed to be continuous and smooth compared to the latter.
- (v) The off-terrain man-made objects usually have a limited size. This characteristic could be utilised for eliminating objects from the DTM as they usually have relatively small area.

However, in some cases ground pixels may not fulfil such characteristics and misclassification errors may occur (Meng et al., 2010). For example, cliffs break with all the aforementioned rules. Another common object type that may incorrectly be classified as ground is bushes, because they have relatively low slope and height difference with their neighbouring ground pixels. Similarly, bridges may also cause misclassification errors since they are smoothly connected to the terrain (Meng et al., 2010).

2.6 Mathematical model for building polygons regularisation

Building model regularisation requires the creation of approximated polygons representing building outlines. Those approximated polygons are usually created by model-driven, data-driven or hybrid methods (He et al., 2014). Modelling of non-rectilinear building polygons is a challenging task due to the difficulties for implementing a set of regularisation rules (Kwak, 2013). The majority of the building footprints can be well-presented by a rectangular shape (Chaudhuri & Samal, 2007) or a set of merged rectangles (Lafarge, Descombes, Zerubia, & Pierrot-Deseilligny,

2008). Thus, the rectilinearity assumption generally serves well for existing methods, while relatively few methods deal with non-rectilinear buildings i.e. (Pohl & Feldmann, 2016).

For rectilinear building polygons, the essential problem of regularisation is how to enforce the orthogonality between the connected polygon edges as well as best fitting of those edges with their corresponding boundary points. Typically, this is should be done by minimising the orthogonal distances between the input boundary points and their corresponding polygon edges in concept of least squares adjustment. This adjustment aims to enhance the approximated building polygons by modelling regular shapes. If those approximated polygons are incorrectly estimated, the adjustment may not necessarily improve the regularisation output. Avbelj (2015) notes that many of the existing approaches skip this step due to the unwelcome additional complexity, and the belief that the possible improvement to the results would only be at a sub-pixel level.

The mathematical model of the least squares adjustment can either be based on the Gauss-Helmert Model (GHM) or the Gauss-Markov Model (GMM) (Avbelj, 2015; Mikhail & Ackermann, 1976). Both models have been introduced for modelling rectilinear building polygons, while an adjustment procedure to deal with non-rectilinear building polygons has not yet been proposed in literature.

2.6.1 Gauss-Markov Model

The Gauss-Markov Model describes a mathematical model between observation equations and unknown parameters. The observation equations are assumed to be linearised initially. Because the functional model is assumed to be linearised, with the observations being a vector l , it has the form (Förstner & Wrobel, 2016):

$$F(\hat{l}) = F(l) + \left. \frac{\partial F}{\partial l} \right|_l (\hat{l} - l) = 0 \quad (2.8)$$

The general form of the observation equations can be written in matrices forms as follows:

$$A \delta + w = e \quad (2.9)$$

$$\text{with } e \sim (0, \sigma_0^2 P^{-1}) \quad (2.10)$$

where, w is a vector of given linearised observation equations; A is the design matrix (also named as *Jacobian* matrix if partial derivatives are included) which has dimension of (number of observations by number of unknowns); δ is a vector of unknown parameters to provide correction to the approximated values; e is a random vector of errors in the observations (which has zero anticipation and $\sigma_0^2 P^{-1}$ dispersion). The term σ_0^2 is known as a prior variance component (constant). The matrix P represents the weight of the observation equations and is symmetric. The inverse of the observation matrix (P) is denoted by Q . The dispersion of e ($D(e)$) is based on the covariance matrix and represented by Σ (positive definite matrix). The relationship between the weight, dispersion, cofactor matrices of the unknown, and the random vector (e) can be written as:

$$D(e) = \sigma_0^2 P^{-1} = \sigma_0^2 Q = \Sigma \quad (2.11)$$

2.6.2 Gauss-Helmert Model

The GH model, also known as *condition equations and parameters* (Avbelj, 2015; Cothren, 2004), is frequently applied for several applications in photogrammetry and geodesy. While the GMM deals with the observation equation and parameters individually, the GHM combines both into a single model. The linearised functional model can be written as:

$$F(\hat{u}, \hat{l}) = F(u^o, l) + \left. \frac{\partial F}{\partial u} \right|_{u^o, l} (\hat{u} - u^o) + \left. \frac{\partial F}{\partial l} \right|_{u^o, l} (\hat{l} - l) = 0 \quad (2.12)$$

where \hat{u} and \hat{l} are the unknowns and the observations respectively. The general form of the observation equations can be written in matrices forms as follows:

$$w = A \delta + B v \quad (2.13)$$

where $A = \frac{\partial F}{\partial u}$ is the design matrix of partial derivatives of the functional model with respect to the unknowns (fixed parameters) at the approximated values u^o ; $\delta = (\hat{u} - u^o)$ is the correction vector to the approximated values; $B = \frac{\partial F}{\partial l}$ is the design matrix of partial derivatives of functional model with respect to the random parameters in the observation equations and $v = (\hat{l} - l)$ is the vector of residuals; $w = F(u^o, l)$ is the misclosure vector. It is assumed that the observation vector is linearised, so that the v vector may be interpreted as the error vector of the observations. This linearised model was presented as the general case of least-squares adjustment by (Helmert, 1872), which is therefore known as Gauss-Helmert Model (Koch, 1999). The stochastic model can then be described as:

$$\Sigma = \sigma_0^2 B P^{-1} B^T = \sigma_0^2 Q \quad (2.14)$$

3 Chapter three Literature review

3.1 Overview

In general, the building footprint extraction process consists of three major steps: DTM extraction, building segments detection, and regularisation. This chapter gives an overview of research literature related to each of these steps.

3.2 DTM extraction

Firstly, in order to generate a DTM the data gaps/holes caused by occlusion or mismatches in the DSM should be filled in advance. Such holes are often filled by an interpolation method (Arefi, 2009; Chaabouni-Chouayakh, Arnau, & Reinartz, 2013).

Complex, data-intensive algorithms have been developed for extracting the DTM. Those algorithms either work on the raw point clouds (Axelsson, 2000; Zhang & Whitman, 2005), or on the gridded DSM (Wack & Wimmer, 2002). Point cloud based algorithms avoid interpolation errors and need less pre-processing. However, these algorithms can be problematic and time-consuming when searching for neighbouring points in margin areas, as illustrated by Meng, Wang, Silván-Cárdenas, and Currit (2009). In contrast, algorithms that work on a gridded DSM can be operated in a similar concept to DTM extraction methods from photogrammetric DSMs utilising image processing tools. These algorithms can overcome problems with neighbouring point searching in margin areas (Meng, Wang, Silván-Cárdenas, et al. (2009) as the neighbours are clearly defined. Additionally, algorithms based on a gridded DSM are computationally efficient to implement utilising available image processing tools.

Established DTM extraction algorithms can be described as slope-based, linear prediction-based, shape-based methods relying on mathematical morphology (Liu, 2008; Mongus et al., 2014; Sithole & Vosselman, 2004) and segment-based (Beumier & Idrissa, 2016). Directional scanning filtering has also proven to be a promising approach for DTM extraction (Meng et al., 2010; Meng, Wang, Silván-Cárdenas, et al., 2009; Y. Mousa, Helmholz, & Belton, 2017; Perko, Raggam, Gutjahr, & Schardt,

2015). All of these methods are based on the physical ground characteristics introduced in Section 2.5. More recently, deep learning methods using the Conventional Neural Networks are also utilised for extracting DTM (Hu and Yuan, 2016).

Several slope-based methods have been developed (Axelsson, 2000; Shan & Sampath, 2005; Sithole, 2001; Vosselman, 2000). Selecting a suitable slope threshold is critical because the terrain slope is not uniform even within the same scene depending on the terrain complexity. Procedures based on adaptive slope thresholds are therefore implemented (Sithole, 2001) to handle such limitation. Promising results can be achieved with these methods in flat areas, while their efficiency declines with increasing terrain slope (Liu, 2008; Mongus et al., 2014).

The linear prediction or interpolation-based methods start by estimating an initial terrain model. Then, height differences or residuals between points and the estimated initial terrain are minimised by linear least-squares interpolation. Negative residuals are given higher weights than positive residuals, assuming that the estimated roughness is usually interpolated over the actual ground surface. Many approaches have been modified based on this concept (Kraus & Pfeifer, 1998; Lee & Younan, 2003). Cubic spline surface minimisation can be also used because of its robustness against outliers (Bulatov et al., 2014; Bulatov & Lavery, 2010). Extraction of detailed terrain surfaces and small objects (not necessarily belonging to the terrain) might be troublesome by the such interpolation methods as noted by Mongus et al. (2014).

Classical morphological filtering is commonly implemented for DTM extraction (Kilian, Haala, & Englich, 1996; Zhang et al., 2003). The method is based on applying morphological operators (i.e. erosion and dilation) (Haralick et al., 1987) in grey-scale images using a structure element. The size of the *SE* is critical to eliminate buildings with diverse size, and therefore, Zhang et al. (2003) proposed a progressive filtering by gradually increasing the sizes of the *SE*. Local elevation difference and slope within the size of the *SE* were applied to identify ground points. Q. Chen, Gong, Baldocchi, and Xie (2007) applied a similar approach, but used an adaptive slope threshold.

Percentile rank filters (e.g. 5%, 10%, 20%, and 40%) were used to mitigate the outlier effects. More recently, Mongus et al. (2014) also applied a morphological filtering approach for building detection purposes. Firstly, laser scanned points were transformed into a gridded DSM. Then, the DSM was smoothed using a Gaussian filter as a pre-processing step. A roughly estimated DTM was then created based on morphological operators. Next, the input laser scanned points were classified as ground points based on their elevation differences from the initially estimated DTM and the slope gradient.

Utilising LiDAR data Zhang and Whitman (2005) tested a slope-based approach, a progressive morphological approach, and a height-based approach using progressively changing window size on coastal and urban regions. The experimental results have shown that the three approaches perform well in urban sites while morphological approach yielded a better result in coastal area. The terrain slope is the most sensitive parameter in all these approaches.

Arefi (2009) proposed a morphological DTM generation method named Geodesic Dilation based on a gridded DSM. Two equal size images are required called mask (J) and marker image (I). The greyscale value of the marker image should be less than or equal to the greyscale value of the mask. In the first iteration, the mask image has the same value as the original DSM. The values of marker image are equal to the minimum value from the original DSM except for the border which is initialised with the same values as in the DSM. The algorithm works by performing four directional scanlines diagonally in two opposite ways. Elevation differences between the minimum and maximum height values within a 3 by 3 pixels sized window along the conducted scanline are calculated. A pixel is labelled as non-ground if its elevation exceeds a certain threshold. The chosen window size is critical for identifying pixels belonging to large buildings (Meng et al., 2010). Nevertheless, building objects located on the border might be classified incorrectly as ground regions in this approach because the height difference might be zero as the mask and marker have the same value (Mousa et al., 2017).

Morphological filtering is a method used to estimate initial terrain surface (Chen et al., 2007) which is then used to identify ground points using elevation difference and slope thresholds. Chen et al. (2007) stated that keeping the terrain surface unchanged when using large *SE* size can prove difficult. Therefore, this approximating process as well as the smoothing step (Mongus et al. (2014)) can both be major sources for possible inaccuracies. In addition, it is anticipated to obtain an over smoothed DTM fluctuated up and/or down the actual terrain depending on the density of the area of interest, *SE* dimension, and the used percentiles (Mousa et al., 2017).

Directional scanning filters have attracted attention in the literature. Meng, Wang, Silván-Cárdenas, et al. (2009) proposed a Multi-directional Ground Filtering (MGF) approach based on LiDAR DSM. The basic idea is to apply scan-lines filtering from left-to-right and from right-to-left within a moving window of pre-defined dimensions. Firstly, the pixel having the minimum elevation value within this window is determined to be a bare ground pixel. If the height difference between the pixel under examination and the minimum exceeds a pre-defined height threshold, the pixel is categorised as a non-terrain pixel. Otherwise, the slope between the current pixel and the following pixel in the considered scanline direction is calculated. If it is higher than a pre-selected slope threshold, it is also considered as a non-terrain pixel. If the slope is less than the slope threshold and positive, the pixel is given the label of the previous pixel. If the slope is non-positive, the distance to the nearby terrain pixel is used to decide a classification (e.g. terrain or non-terrain pixel). Meng, Wang, Silván-Cárdenas, et al. (2009) conducted a comparative study with eight different algorithms. The results proved the robustness of their approach.

Similar to (Meng, Wang, Silván-Cárdenas, et al., 2009), Perko et al. (2015) developed a directional scanning approach called the Multi-directional and Slope Dependent (MSD) method for DTM extraction from DSM derived from satellite imagery. Points are evaluated based on height differences and the slope from eight directional scanlines. The slopes of the current point under examination, and its neighbours within a 3 by 3 pixels sized window are calculated for each point. Unlike the MGF algorithm, the local terrain slope is considered in this algorithm. This is similar to the approach

proposed by Piltz, Bayer, and Poznanska (2016), but they generated an off-terrain mask including buildings and trees using a metric called Normalised Volume above Ground (NVAG) which is then used for DTM creation.

Segment-based methods involve three main stages, including segmentation of laser scanned points or gridded DSM, labelling clustering segments into terrain and off-terrain groups, and removing off-terrain segments and filling gaps by an interpolation technique (Beumier & Idrissa, 2016). This segmentation procedure aims to classify points having similar features into groups based on their homogeneity (Tóvári & Pfeifer, 2005), such as roof facets as an example. Segmentation of raw laser scanned points may use profiles or region growing procedures while segmentation of the DSM is usually based on the slope calculation (Gevaert, Persello, Nex, & Vosselman, 2018). Region growing procedures usually utilise the geometric properties of a point in relation to its neighbours, such as elevation or slope (Tóvári & Pfeifer, 2005).

Sithole and Vosselman (2005) segmented laser scanned points based on the continuity characteristics of the terrain. Profiles created by the scanline segmentation procedure are examined in multiple directions. Points in those profiles were analysed in order to perform classification. For instance, points belonging to the terrain are assumed to be continuous and smooth, and conversely, object points are assumed to be discontinuous. Tóvári and Pfeifer (2005) combined a segmentation procedure based on region-growing of laser points with DSM filtering. For instance, the normal vector of each individual laser point is estimated using the k nearest neighbours. Then, points were grouped into segments according to the similarity between the estimated normals. The DSM was interpolated from the laser points using least squares to fit a first order polynomial. Finally, height differences between the interpolated DSM and the segmented points were analysed. For example, points having height differences in the range from -2 meters to +2 meters were assumed to be ground points.

In relation to the segmentation of the DSM, Hingee, Caccetta, Caccetta, Wu, and Devereaux (2016) utilised the gradient direction, area, and slope parameters to identify candidate terrain segments. The DSM was then smoothed using a surface fitting

method. Similar to Tóvári and Pfeifer (2005), height differences between the smoothed DSM and the terrain segment candidates were analysed to improve the quality of those candidates by removing segments having elevation differences larger than a user-defined threshold.

Beumier and Idrissa (2016) proposed a segmentation procedure using a photogrammetric DSM. The input DSM was first smoothed using a mean-shift filter, then segmentation, followed by regions-filtering. The process is then repeated. This segmentation procedure was implemented to separate the DSM into regions based on analysis of the height information within a certain window. The region filtering was applied for rejection of locally higher parts, which usually related to off-terrain objects (such as buildings) depending on neighbourhood analysis. The remaining regions were normally matched to roads, large terrain surfaces, or fields. Finally, data holes resulting from rejecting non-ground objects in the previous step was then interpolated using a bilinear interpolation technique to generate the final DTM.

In the same context, Yan, Blaschke, Liu, and Wu (2012) and Hingee et al. (2016) proposed a segmentation procedure considering the slope measurement and neighbourhood analysis to distinguish between terrain segment and off-terrain segment. Over- or under-segmentation is the main limitation of such algorithms (Tóvári & Pfeifer, 2005). Difficulties in automation of the implemented parameters are noticeable (Pérez-García, Delgado, Cardenal, Colomo, & Ureña, 2012) due to the large variation of the terrain types. For instance, the slope values can significantly vary between adjacent points in flat terrain compared to those in sloped terrain.

Machine learning approaches have also attracted attention using the convolutional neural networks (CNN) (Long et al., 2015). While, the main purpose is the image classification, i.e. buildings, trees, road et cetera some methods are aimed at extracting the DTM (Hu & Yuan, 2016). The CNN method aims to implement a training system for the purpose of object classification in cooperation with a supervised classifier in an end-to-end procedure (Marcos, Volpi, Kellenberger, & Tuia, 2018). Accordingly, these deep learning methods have been applied to extract the DTM from imagery data

(Gevaert et al., 2018) or from LiDAR data (Hui et al., 2019). LiDAR data must be transformed into a raster format first as the CNN methods work based on image processing tools. Its advantage is in its effective image classification at pixel-level accuracy. The main disadvantage, however, is the expensive training effort required (Mousa et al., 2017).

There is general consensus in the literature that the slope is the most sensitive parameter in the processing of DTM generation. For instance, Mongus et al. (2014) mentioned that slope-based approaches have difficulties in sloped terrains (i.e. significant variations in heights). In addition, defining a slope threshold in terms of terrain information in the analysed scene is, to some extent, subjective (Zhang et al., 2003). This problem becomes obvious in high-resolution DSMs because the slope value between the neighbouring pixels is considerable without transitioning from ground to non-ground, or vice versa (Mousa et al., 2017). Therefore, eliminating the slope parameter from the processing could help to improve the quality of the generated DTM.

3.3 Building detection in terms of data sources

Building detection relates to identifying building object regions, where they are located, and distinguishing them from other non-ground objects such as trees. There are numerous existing approaches which can be categorised into three groups in term of their respective data sources, including images, LiDAR data, and approaches that combine both images and LiDAR data.

3.3.1 Building detection from 2D image data only

Approaches based on images only commonly implement segmentation and classification procedures. For example, Müller and Zaum (2005) proposed a segmentation procedure based on region-growing. Firstly, they used airborne colour images converted to grey scale and applied the linear regression classifier to create the initial segments/seeding areas. Then, the region-growing algorithm was applied to the

entire image utilising geometric (e.g. size, area), photometric (e.g. mean hue), and structural (e.g. neighbourhood and shadow) properties. Building segments were detected based on those properties. Demir (2013), claimed that this approach performs well with only red roof buildings.

Using only imagery information is still problematic because of the similarity of spectral properties between buildings and other man-made structures like roads (Chaabouni-Chouayakh et al., 2013; Sirmacek, d'Angelo, & Reinartz, 2010; Tian, Chaabouni-Chouayakh, & Reinartz, 2011). In addition, the variety of the man-made objects and their occurrence mitigate the classification efficiency (Beumier & Idrissa, 2016) especially with very high resolution images (Cheng, Gong, Chen, & Han, 2008). Therefore, and due to the success that has been achieved by approaches based on DSMs (Haala & Kada, 2010), integration of both inputs were considered in the most relevant studies.

3.3.2 Building detection from 2D and 3D image data

Several methods have been developed to detect buildings from a combination of image data and DSMs. For example, a method that utilised a DSM with optical satellite imagery was proposed by Ozcan et al. (2013). Firstly, the Harris corner detection technique was applied to detect corner points located in the input images. Presence of such points is an indicator for building locations. These corners were then fused with the DSM data. Vector directions are generated from the extracted corners towards the building centre. Lastly, a kernel density map was generated which indicates the estimated building locations using a symmetric Gaussian probability density function.

Tian and Reinartz (2013) used panchromatic and multispectral images obtained from the WorldView-2 sensor, in conjunction with a DSM. Firstly, a DTM was generated using morphological filtering, which was then subtracted from the DSM to create the nDSM. Trees were removed using an NDVI mask in order to estimate building locations. A Hough transformation is then applied on panchromatic images to extract building boundaries via line segments. In addition, a random forest (Breiman, 2001)

is applied on the multispectral images to extract the building locations. Finally, the estimated building locations and the panchromatic image segmentation results, as well as the random forest classification results, are fused together to extract the building outlines.

Qin and Fang (2014) proposed a hierarchical approach using aerial images and photogrammetric generated DSM. The first step is to distinguished shadow regions using a morphological index in order to effectively perform calculation of NDVI. Initial building locations are estimated by a combination of morphology top-hat extraction of the DSM and the NDVI scheme. Finally, a graph cut optimization based on pixel segmentation was modified to integrate the DSM and the multispectral information to estimate building locations.

More recently, the Convolutional Neural Networks (CNNs) (Long et al., 2015) have proved to provide effective methods for comprehensive image classification (i.e. building, trees, grass, and bare ground). It works by building a parametric training system for identifying man-made objects jointly with a classifier (Marcos et al., 2018) in an end-to-end method. Its advantage is in its effectiveness of image classification at pixel-level accuracy. As mentioned previously, an expensive training process may be required. Nevertheless, depth data (i.e. nDSM) seems to be an essential input in the processing, which can be created after extracting the DTM (only). Many recently reported methods have combined depth information with CNNs (Griffiths & Boehm, 2019; Marmanis et al., 2018; Piramanayagam, Saber, Schwartzkopf, & Koehler, 2018).

3.3.3 Building detection from LiDAR data only

LiDAR sensors offer highly reliable and accurate 3D surface measurements. However, the raw 3D point cloud data are randomly distributed and need to be organised in such a way that valuable information such as buildings can be extracted. Interpolating onto gridded data is the preferable option (Ma, 2004). Since LiDAR data is comprised of primarily position and height observations, all the existing algorithms are based on the

fact that buildings should have a significant height difference from their surrounding area. Hence, extracting a DTM is a key step for building detection (Mongus et al., 2014), and it must be computed first (Bulatov et al., 2014).

Within approaches that utilise the 3D point cloud data the segmentation approach is the most common. This segmentation aims to group points belonging to the same class based on analysis of the height and slope for a certain point with its neighbours. Generally, two common segmentation procedures can be found in the literature, region-growing (Awrangjeb et al., 2014; Sampath & Shan, 2007) and clustering in the attributes space (Filin & Pfeifer, 2006; Kwak & Habib, 2014; Lari, Habib, & Kwak, 2011).

The region growing procedures (Vosselman, Gorte, Sithole, & Rabbani, 2004) start with identifying seed points that belong to the terrain. Then, the algorithm starts collecting adjacent points based on pre-selected parameters, such as height differences and slope to distinguish ground from non-ground points. The same concept can be also applied on a LiDAR DSM (Chen, Teo, Shao, Lai, & Rau, 2004; Geibel & Stilla, 2000; Jarzabek-Rychard, 2012). Rule-based procedures (e.g. area and shape) were considered to distinguish between building and trees (Awrangjeb et al., 2014).

The second category of approaches (i.e. clustering in the attributes space) starts with computing attributes, for example, position of the local normal projection on the derived best fitted surface for each laser scanned point (Lari et al., 2011). A clustering step is then performed in order to identify homogenous points according to these attributes. The clustered segments are analysed based on some criteria (i.e. slope discontinuity, area, and height difference) to classify whether a point belongs to a ground or non-ground segment. Rule-based approaches can be applied to remove trees (Kwak & Habib, 2014). The main difference between these two approaches is that the region growing is applied to the spatial domain, while the clustering approach is applied to the attributes space (Kwak & Habib, 2014). The quality of both procedures depends on the quality of the selected seed points and the computed attributes. The

main limitation of solely point cloud methods is of the inability to remove trees completely (Yoon et al., 2008).

Some building detection methods utilised laser points with interpolated DSMs. For instance, Meng, Wang, and Currit (2009) identified man-made objects from ground objects by filtering LiDAR-DSM. Then, the multi return properties of the laser points were utilised for the purpose of removing trees, but the ground vegetation could not be identified in this method.

Ma (2004) proposed a texture information method, based on analysing the height difference of a point and its neighbours in a predefined window size. The differences between points and the plane in this window are utilised. Pixels located in a non-plane surface are eliminated because they are most likely to belong to trees. Zhang, Yan, and Chen (2006) used a similar approach to eliminate trees. Both approaches were not able to remove all trees, hence, a rule based procedure with respect to area was applied to remove the remaining trees.

Other attempts have also exploited the geometric properties of trees such as area, shape, and compactness (Awrangjeb et al., 2014; Meng, Wang, & Currit, 2009) or surface roughness (Pfeifer, Rutzinger, Rottensteiner, Muecke, & Hollaus, 2007). Such procedures are able to remove separated trees, however suppressing those trees touching buildings is problematic. Distinguishing buildings from trees is a challenging task even in the case that a Normalised Difference Vegetation Index image is available (Markus Gerke & Xiao, 2014), especially in shadow regions.

3.3.4 Fusion of LiDAR data with imagery-derived information

There are two ways data fusion of LiDAR 3D point cloud and imagery data is commonly used. Firstly, the nDSM is generated from LiDAR data, while the NDVI imagery is used only for tree removal. This can lead to poor building planimetric accuracy (Awrangjeb et al., 2010). Alternatively, extra imagery information like line segments can be exploited to improve the geometric quality of the building footprints.

However, buildings or building-parts located in shadow area can be missed completely (Awrangjeb et al., 2010; Qin & Fang, 2014; Rottensteiner, Trinder, Clode, & Kubik, 2005).

Due to the fact that trees may be difficult to be eliminate using LiDAR data only, several algorithms have used an NDVI image for this purpose (Huang & Zhang, 2011; Rottensteiner et al., 2007; Sohn & Dowman, 2007). While there are different sources of fusion errors (e.g. misalignment and resampling data) (Meng, Wang, & Currit, 2009), this option has been the trend for the majority of approaches. However, the inefficiency of distinguishing trees located in shadow regions is the main drawbacks of using the NDVI.

In order to mitigate the uncertainty of the NDVI, Rottensteiner et al. (2005) and (2007) computed strength metrics of the surface roughness and directness based on the concept of Förstner (1994) as texture information from LiDAR-DSM. The strength means the consistency measure of polymorphic feature detection performed on the first derivatives of the DSM. It represents the smoothed squared sum of the second derivatives of the DSM inside a defined filter size. Strength and directness were utilised to detect trees assuming that trees have less homogeneous surface compared with buildings. The elevation difference between two DSMs generated from the first and last LiDAR pulse returns were also exploited as a measure for tree removal.

Other challenges of image analysis relate to the diversity of tree colour, and density of leaves due to changes in seasons. Awrangjeb, Zhang, and Fraser (2012) combined texture information called entropy with NDVI to mitigate such problems. It has been suggested that the fusion of compactness and NDVI may mitigate these effects (Gerke & Xiao, 2014). However, this may not help for removing trees hugging buildings because both are merged in one segment.

More recently, the International Society of Photogrammetry and Remote Sensing (ISPRS) has made available a benchmark dataset including airborne images, LiDAR data, DSMs, and orthophoto for two cities (Vaihingen in Germany and Toronto in

Canada) for the purpose of urban object detection (Cramer, 2010). Many participants developed and tested their methods on the data provided. Their results were presented and evaluated in (Rottensteiner et al., 2014). All the methods showed a satisfactory result dealing with buildings larger than 50 m² in Vaihingen, while no method reported could detect all the larger buildings in Toronto. The authors expected that the reason might be related to those complex buildings having extremely diverse elevation levels, which lead to missing some buildings having relatively low heights. Detecting small buildings of less than 50 m² proved difficult. Many of them were missed by all methods, especially in Toronto. In general, methods based on a LiDAR DSM and orthophoto have shown higher average area-based quality than those based on point clouds and original images (Rottensteiner et al., 2014).

In spite of the numerous approaches developed for building detection, the problem has still not been fully solved yet, and more work has to be done. For instance, problems related to eliminate trees (Gerke & Xiao, 2014), others are related to the quality of the generated DTM as presented in Rottensteiner et al. (2005), or are related to the complexity of the building objects (Rottensteiner et al., 2014).

3.4 Approximate building polygons generation and refinement

The result from the previous step is either a building mask (e.g. a binary image), or segmented laser points. In both cases, a list of boundary points is required for further enhancement. Those boundary points can be identified, for instance, using the Moore contour-tracing algorithm (Gonzalez, Eddins, & Woods, 2004), convex-hull-based technique (Pohl et al., 2017), or alpha-shape method (Albers, Kada, & Wichmann, 2016). This list of points represents a vector chain defining the building outline, usually appearing as a jagged structure, which needs to be regularised for applications such as 2D map generation and 3D city modelling. The number of the boundary points needs to be reduced to the minimum in such a way that the characteristics of the original building shape are preserved. This step is referred to as building footprint extraction, and is followed by a refinement (i.e. simplification) of the building outline. The result should be a polygon describing the same properties of the original building object with a significant reduction of the number of boundary points.

Building polygon generation can be done based on the detected building results (as discussed in 3.3), or utilising additional information such as cadastral maps (Henn, Gröger, Stroh, & Plümer, 2013; Kada & McKinley, 2009; Vallet, Pierrot-Deseilligny, Boldo, & Brédif, 2011). However, these are not always available or maybe not up-to-date (Brédif et al., 2013; Pfeifer et al., 2007).

Methods based on oblique images and video sequences for extracting 2D building outlines have been also attracted attention by researchers (Bulatov et al., 2014; Bulatov & Lavery, 2010). Line segment algorithms (e.g. Canny (1986)) are commonly applied on orthophoto images to enhance the planimetric accuracy of building outlines (Awrangjeb et al., 2010) where only line segments which are more than 3 meters in length are considered. However, generation of an orthophoto is either based on DSM or DTM (Rottensteiner et al., 2014) in order to mitigate the geometric errors caused by the relief displacement. These errors rise with increasing building height and therefore appropriate co-registration of image and DSM is needed to manage these errors (Gerke & Xiao, 2014).

Since a DSM is the reference for creating the true orthophoto, edges or outlines obtained from LiDAR DSM may be more accurate compared to those obtained from ortho-photo depending on the point clouds density. The efficiency of image-based line segments reduces significantly in shadow regions. According to (Kwak, 2013), lines or edges extracted from images do not necessarily belong to the actual building outline because of shadow effects, noise, or non-relevant features close or on the top of the building. Therefore, an appropriate solution based only on DSM might be preferable in order to overcome the co-registration or alignment errors and to avoid the shadow problems. Generally, building 2D outlining can be grouped into model- and data-driven methods (Tarsha-Kurdi, Landes, Grussenmeyer, & Koehl, 2007), or a hybrid model combining both methods (He et al., 2014).

3.4.1 Model-driven approaches

The model-driven approaches provide regular approximated polygons where the data is required to perform a best-fit. The aim is to approximate a parametric model from

an existing library which comprises of basic building designs to be fitted to a given set of boundary points. These are robust and computationally effective approaches. In addition, the constraints and topological relationships between the polygon sides have to be fixed (Avbelj, 2015), which reflect the orthogonality and parallelism assumptions. Therefore, buildings having characteristics not represented in the library of models will be problematic.

The majority of the existing model-driven approaches, (e.g. (Hammoudi & Dornaika, 2010; Henn et al., 2013; Kada & McKinley, 2009) have been developed in terms of 3D building model construction. In the context of the 2D model-driven approaches, the Minimum Bounding Rectangle (MBR) procedure is commonly implemented (Arefi, 2009; Avbelj, 2015; Dutter, 2007; M Gerke, Heipke, & Straub, 2001; Kwak & Habib, 2014). Overcoming the problem of noisy building data is the main advantage of this approach due to the prior assumption of the building shape. The main drawback is that only buildings with right angles can be modelled, which is often not the case for complex buildings. Accordingly, non-rectilinear buildings are erroneously simplified, or represented by overly complex polygons (Avbelj, 2015).

The Minimum Bounding Rectangle can be estimated from given boundary of points (Figure 3.1(b)) representing a building segment (Figure 3.1(a)), which then, used as an initial solution (the red polygon in Figure 3.1(c)). The MBR is computed by rotating the original polygon around its centroid sequentially (Arefi, 2009). The bounding box with minimum area is the MBR, with consideration given to the main direction. If more details are required, the building segment is subtracted from the bounding rectangle, and a new rectangle is found from the remainder if it has a significant area, as can be seen in the black object in Figure 3.1 (d). The part represented by the green polygon (Figure 3.1 (f) should be eliminated from the MBR as shown by the red polygon (Figure 3.1 (g)). Nevertheless, sometimes, the orientation of the sub rectangles does not align with the orientation of the first MBR, depending on the shape of the obtained overlapped area (the black area in Figure 3.1(d). If the new rectangle has an overlapping region with the building segment, this overlapping region is added. The procedure of subtracting and adding rectangles is recursive and is repeated until

the deviation between the building mask and the rasterised polygon is smaller than a user-defined threshold. Thus, it is denoted as the Recursive MBR (RMBR). More details can be found in Avbelj (2015) and Kwak and Habib (2014).

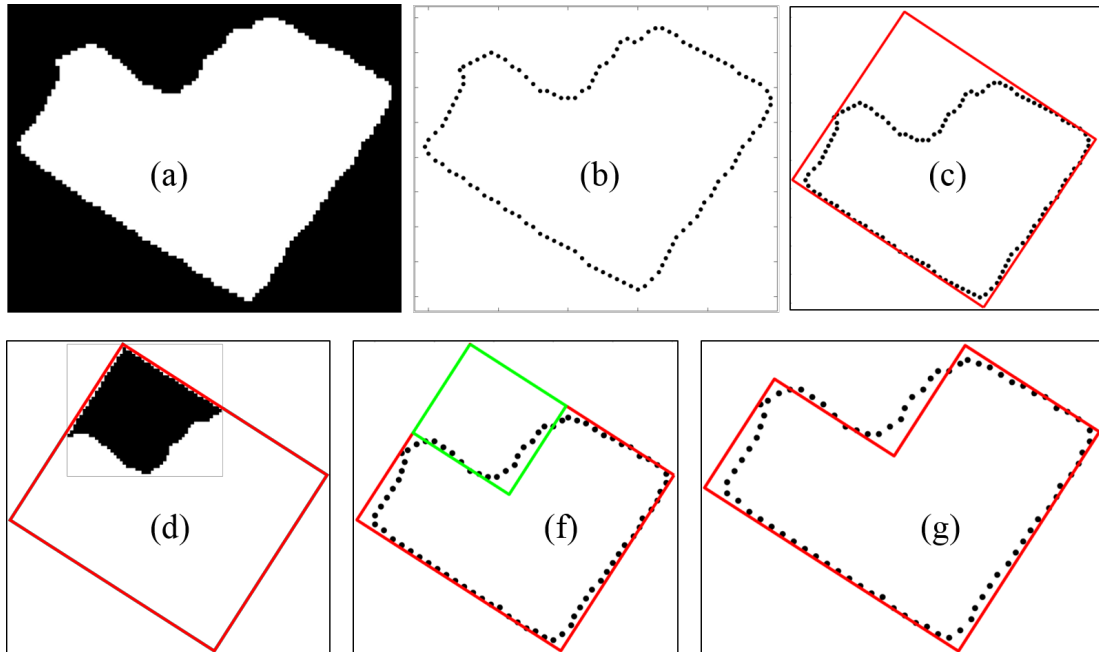


Figure 3.1: Procedure of generating RMBR. Raster image shows building segment (a). The extracted boundary points (black dots) are shown in (b). (c) shows the MBR created (red box). Non-overlapped area (black) resulting from the subtraction of the original building segment from the MBR is given in (d). (e) shows the new polygon (green) which should be extracted from the MBR. The final RMBR extracted is given in (g).

Based only on a DSM, Lafarge et al. (2008) proposed a model-driven approach using Marked Point Process (MPP). The authors make the assumption that buildings can be described by multiple connected rectangles. Connected rectangles are delineated by enhancing the connection between the corresponding edge of the roof, and the neighbouring rectangles. The resulting footprints are then exploited in the 3D building reconstruction. A similar approach has been presented by (Brédif et al., 2013).

3.4.2 Data-driven approaches

In contrast to the model-driven approach, existing knowledge of the building model is not required in the data-driven methods. They are free for pre-existing constraints, and therefore offering more flexibility in formulating any building outline. However, data-driven methods still face difficulties in the case of missing data to be recovered (Sohn et al., 2012). Additionally, it is often difficult to formulate constraints or a set of rules to implement the data-driven methods, making them more complex than the model-driven ones (Kwak & Habib, 2014).

The data-driven approaches commonly incorporate a combination of the Douglas-Peucker (DP) algorithm (Douglas & Peucker, 1973), the Random Sampling Consensus (RANSAC) method (Fischler & Bolles, 1981), and the Hough transform (Hough, 1962). A smaller number of reported studies are based on histogram analysis (Alharthy & Bethel, 2002; Pohl et al., 2017) or on the Sleeve line fitting algorithm (Zhiyuan Zhao & Saalfeld, 1997); Zongze Zhao, Duan, Zhang, and Cao (2016)).

The Douglas-Peucker (DP) algorithm has been acknowledged by Jwa et al. (2008) as the most effective simplification algorithm of open and closed polygonal chains. Starting from an arbitrary point (the red triangle in Figure 3.2(a)), the algorithm connects a straight line (the blue line in the same figure), which divides the boundary points into two approximately equal sub-chains. Then, a new polygon vertex is determined (the green star indicated by the blue arrow in Figure 3.2(a)), which corresponds to a boundary point that has the maximum distance to the blue line. This is then utilised to obtain the new nominated edges represented by the blue line in Figure 3.2(b). The procedure recursively extracts straight lines from the recently marked vertex, which has a distance to the determined edges larger than a pre-defined tolerance until returning to the start point. Figure 3.2(d) shows the final simplification result depicted by the green closed polygon.

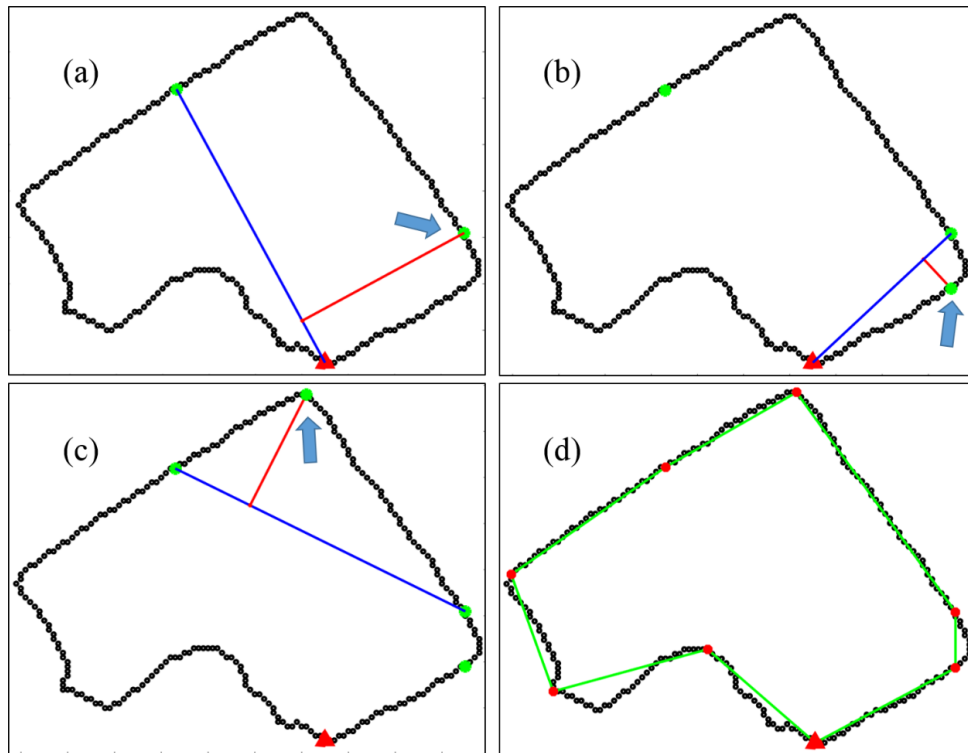


Figure 3.2: Concept of the DP algorithm. (a) shows the first conducted line (blue) starts from the first vertex (red triangle) and divides the boundary points (black dots) into two sub-chains, while the red line nominates a new vertex corresponding to the farthest boundary points to the blue line. The algorithm iteratively creates straight lines from new nominated vertices that has farthest distance to the nominated edges until returning to the start point as shown in (b) and (c). (d) shows the final created polygon (green).

The Hough transform method is also used to extract line segments identified by a parametric equation including boundary point coordinates, distances to the origin, and direction of the line normal (angle). Each boundary point is converted into a curve in the parameter space. Then, it uses an accumulator array to accumulate votes for the parameters of each line segment. The result is a set of non-connected and overlapped line segments corresponding to the most plausible lines that can be detected based on the parameters. Further post-processing is required to obtain a closed polygon, including merging, extending, and deleting. A number of studies are based on this algorithm (e.g. Albers et al. (2016), Guercke and Sester (2011), Höhle (2017)) and (Widyaningrum, Gorte, & Lindenbergh, 2019). Line segments shorter than 1.5 metres

were not considered in Albers et al. (2016). It is also useful to detect edges which are parallel or perpendicular to each other (Arefi, 2009), as well as to find the dominant building direction. The disadvantages of Hough transform is that it is computationally inefficient and sensitive to the fitting parameters (Mongus et al., 2014).

The RANSAC algorithm is a robust line fitting algorithm that is often applied to data with a high level of outliers. A random number of points have to be selected first (minimum two points in the case of lines) to solve for the model (straight line). A distance threshold is utilised to identify points as inliers (points having a small distance to the fitted line), otherwise they considered as outliers. Compared to the DP algorithm, this distance threshold has a similar meaning as the tolerance threshold. If the number of inliers is higher than a certain defined value, the fitted line is stored and the inliers points belonging to it, are identified. A different set of random points are then chosen and the process is repeated. After a specific number of iterations, the model with the highest consensus group (largest number of inliers points) is determined and the model computation is repeated using all the inlier points to fit the line. The process is repeated until no further lines can be extracted. Similar to the Hough transform, the output is a set of disconnected line segments. Therefore, the adjacent line segments need to be connected, and then reoriented based on the rectilinearity assumption followed by an adjustment step (Jarząbek-Rychard, 2012; Neidhart & Sester, 2008; Sester & Neidhart, 2008).

The aim of the RANSAC and Hough algorithms is to detect line segments. This does not guarantee that enough lines will be detected in order to form a building outline. Therefore, further processing (e.g. extending, merging, deleting) is required in order to obtain a closed polygon that can be used as an initial approximation for a building footprint. If a significant part of the building is missing from the data, it is very difficult to recover it again. In addition, it has been shown that the so called multi-model RANSAC is very problematic as the number of the models needs to be identified by the user (Toldo & Fusiello, 2008). Most of the existing data-driven approaches have been implemented so that they exploit an initial solution based on the DP algorithm (He et al., 2014; Jwa et al., 2008; Maas & Vosselman, 1999; Sohn et al., 2012; Wang

et al., 2006). This is because it can provide closed polygons directly without further processing. It has been empirically demonstrated that the DP is an accurate line segment algorithm (Shi & Cheung, 2006) that is able to maintain the original shape (Song & Miao, 2016). Thus, many existing methods have utilised it e.g. (Li, Rottensteiner, & Heipke, 2019)

In terms of the model-driven approaches (e.g. RMBR), they are designed for the case where buildings are comprised of right angles only between lines. When dealing with a complex building structure, such assumption are not suitable (He et al., 2014). For modelling regular shapes with a low level of noise the DP model is often sufficient and generates acceptable results. However, the efficiency of the DP model declines when increasing the irregularity of the building boundary points, because it is inherently sensitive to the noise and to the position of the starting point (Mousa et al., 2019).

In data-driven approaches, building characteristics like angle detection and area preservation are not considered in the processing. This inherent flexibility of data-driven approaches is considered preferable to solve the problem of modelling the more complex buildings. Further, even though the obtained polygons from model-driven methods (e.g. RMBR), are smoother (more regular) than those generated by data-driven methods (e.g. the DP algorithm), refinement of both model-driven and data-driven outputs are necessary. This refinement is most performed in terms of the least-squares adjustment.

3.4.3 Refinement of building polygons

The goal of refining the building polygons is to find the best-fit polygon edges to the given boundary points (in this context called observations) using optional weighting and constraints which enforce the geometric relationships between the polygon edges. Because the building footprints can usually be well represented by a rectangular shape or a set of joint rectangles, they are often assumed to be modelled by rectilinear polygons. Hence, the orthogonality and parallelism are the basic assumptions in all of

the existing methods. Since the adjacent polygon edges are orthogonal, the parameters of such edges can be estimated, and the mathematical model can be written as a slope function (Sampath & Shan, 2007), or normal function (Avbelj, 2015; Kwak & Habib, 2014).

For rectilinear polygons, constraints for the edge or line segment are required (e.g. location, rotation, height and width, or normal vector distances between each edge and the origin of the coordinate system), and the relation between the adjacent edges (Avbelj, 2015). In some studies, the longest edges are given more weight in the refinement adjustment (Sampath & Shan, 2007), or the weights are estimated from the spectral imagery properties (Avbelj, 2015). In comparison, others perform the adjustment without weighting (Awrangjeb et al., 2010; Kwak & Habib, 2014). Partovi, Bahmanyar, Krauß, and Reinartz (2017) claimed that giving different weights does not influence the accuracy of resulting polygons and therefore, the identity matrix was used.

According to Avbelj (2015), while adjusting the detected line segments individually, the topological relationships of edges will be lost and therefore geometric constraints should be included in the adjustment by an extra set of Lagrangian multipliers or pseudo observations. For example, Arefi (2009) estimated the dominant building direction based on analysing line segments derived by Hough transform. Then, the other edges of the polygon were adjusted accordingly. While, Awrangjeb et al. (2010) and Awrangjeb (2016) assumed that the main building direction corresponds to the longest line segment which is then used as the basis to adjust the other building edges. Sampath and Shan (2007) considered that the longest line segment most likely corresponds to the main direction. Then, the extracted edges were grouped into two observation groups which were either parallel or perpendicular to the main direction to achieve the adjustment. Kwak and Habib (2014) defined the main direction from the derived MBR (with a significant overlap with the boundary points) after adjusting it. Such MBR with this condition is more reliable, but the method of how to estimate it was not defined. The subsequent rectangles were adjusted according to orientation of the first MBR. Another adjustment procedure proposed by Höhle (2017) constrains

the adjacent line segments to be orthogonal. Coordinates of corner points are estimated first. During adjustment, the positions of corners are updated until every two neighbouring line segments are orthogonal.

Solving the adjustment of rectilinear polygons requires two steps. Firstly, solving the orthogonality problem as described above. The second step is to solve the best-fit of the polygon edges with their corresponding boundary points, which can be achieved by minimising the orthogonal distances between them. The majority of the existing methods exclude the second step. Nevertheless, these two steps must be solved simultaneously in order to achieve more accurate building polygons.

Determining the dominant building direction is a challenging task, especially for rectilinear buildings. In the literature two approaches have been found when estimating an initial solution. The first approach is the estimation of the main direction based on methods such as Hough transform (Arefi, 2009) or the longest line segment (Awrangjeb et al., 2010), where it is fixed and the other polygon edges are adjusted accordingly. The second assumption is the estimation of the dominant direction from the derived MBR (Avbelj, 2015; Kwak & Habib, 2014), which is then used to adjust the polygon accordingly. In both cases, the initial solution for the dominant direction is not necessarily accurate enough to provide an acceptable solution.

Based on the initial solution, boundary points are labelled to belong to their corresponding polygon edges. However, none of the existing building outline regularisation methods proposes re-labelling of the boundary points to their corresponding rationalised edges during the adjustment. Consequently, the accuracy of the resulting building polygons is limited to the initial solution, and further improvement in the geometric accuracy may not be achieved. Therefore, a method for updating the assignment of the boundary points in the adjustment is recommended to enable greater regularisation accuracy.

4 Chapter four Methodology

4.1 Overview

This chapter details the proposed building footprint extraction methodology, as summarised in Figure 4.1. Initially the LiDAR point cloud data are cleaned from outliers, then rasterised as a pre-processing step. A new DTM extraction algorithm is then proposed to overcome shortcomings of the existing methods. Further, nDSM, NDVI, and the planarity map are generated and fused for the purpose of detecting building regions. The approximate building polygon is generated. This step includes a novel data-driven method to create approximate polygons for rectilinear and non-rectilinear buildings. Boundary refinement based on the adjustment of a GMM and GHM is performed. A new algorithm is introduced for assigning boundary points to their corresponding polygon edges. Accordingly, a new procedure for calculating the absolute accuracy (similarity) between two polygons (i.e. extracted polygon and its reference) is explained utilising the RMSE measure.

4.2 Pre-processing

Points of a 3D LiDAR data are randomly distributed in space, meaning they are not ordered in a regular grid pattern due to the outliers. Transforming them into a regularly gridded DSM is needed for computationally efficient post-processing. To do so, outlier removal must be performed in advance of this process. The next two sections describe both processes.

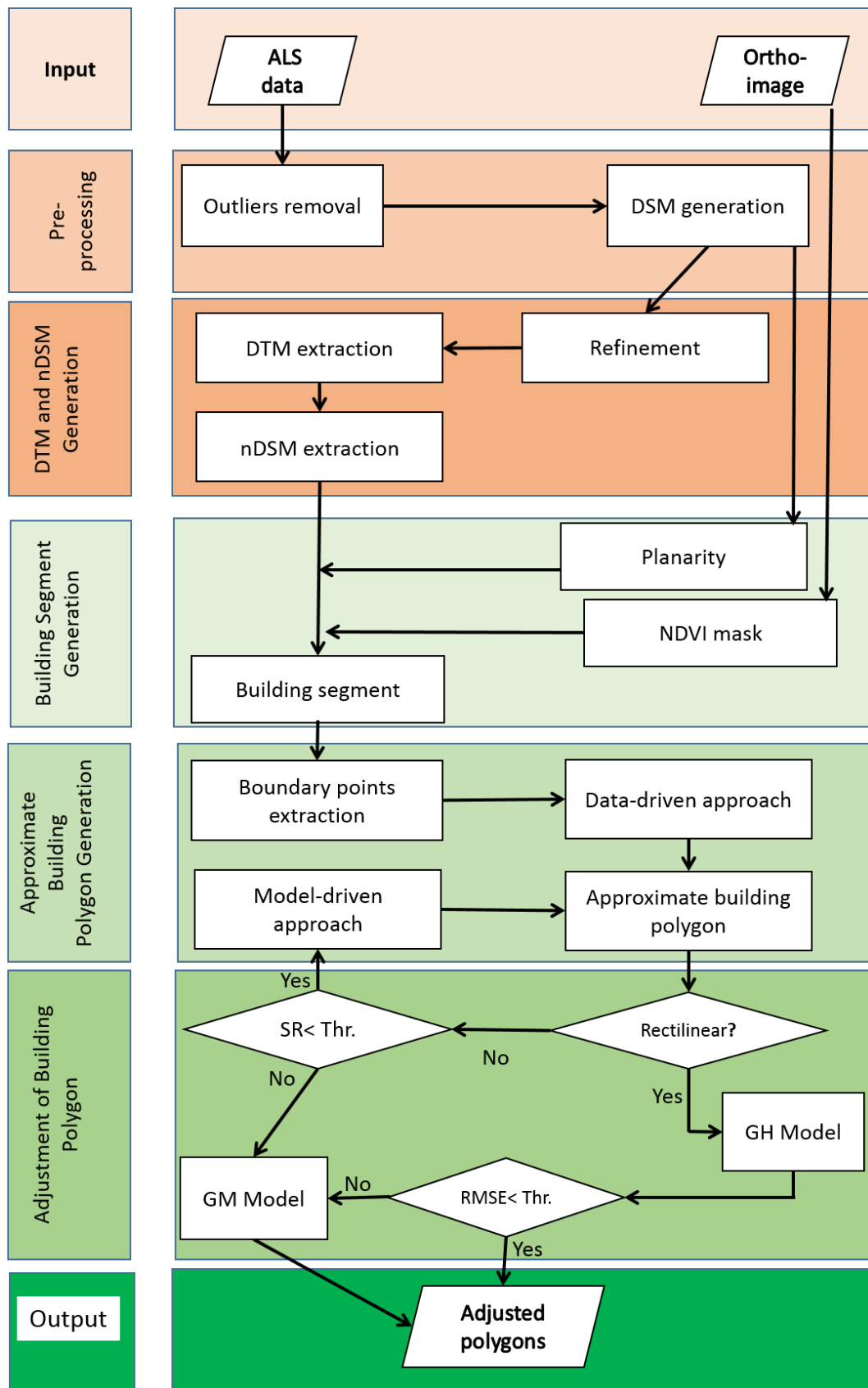


Figure 4.1: Workflow of the proposed methodology.

4.2.1 Outliers removal

Laser scanned points commonly contain outliers. Two categories of outliers may exist, namely low outliers (e.g. negative elevation values) and high outliers (e.g. too high elevation). Low outliers are usually caused by reflection of light from elements in the atmosphere, power lines, or birds. High outliers are mainly caused by multiple reflections. The type of the target material plays an important role on light reflection, such as water bodies. Consequently, the reflected laser light can be too weak or never observed by the receiver leading to missing data.

The most common statistical method for outliers detection is based on the normal distribution of the data, e.g. Grubbs' Test (Grubbs, 1969). In this thesis, the procedure is applied based on the normal distribution as well as visual inspection. The first step is to visualise the normal distribution of elevation data (Z), as shown in Figure 4.2(a). In this example more than 99 % of the data is distributed between zero and 70 metres, even though there are isolated raw data points with elevations as low as negative 300 m, and as high as 300 m. Thus, the low and upper limits LL and UL quantiles of Z are computed as:

$$(LL, UL) = \text{quantile}(Z, [a, b]), \quad (4.1)$$

where a and b are the cumulative probability values e.g. [0.0001, 0.9999]. Points that have an elevation lower than LL or higher than UL are omitted.

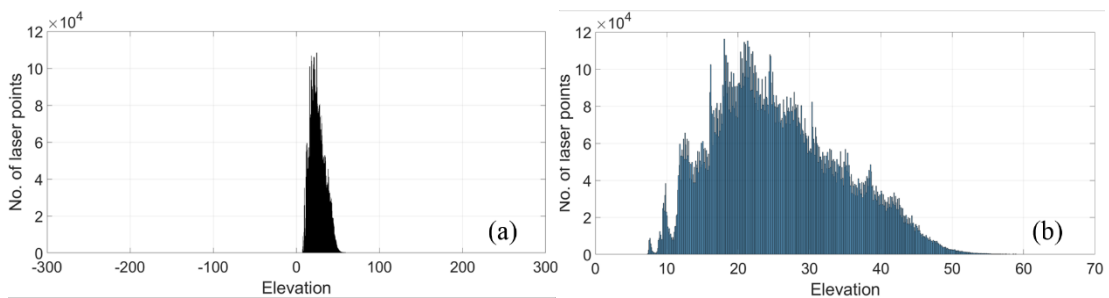


Figure 4.2: Normal distribution of data, (a) and (b) before and after outliers removal respectively.

The outliers were removed in the first stage for all input point clouds in the case study area. Accordingly, the normal distribution of the data after removing the outliers is shown in Figure 4.2(b). However, if the data has many outliers (i.e. locally and

globally) this procedure may not be able to remove the outliers in an efficient manner. In this case data can be divided into smaller patches and the same process is then performed individually on all the patches as suggested by Carrilho, Galo, and Santos (2018).

4.2.2 Data rasterisation

The 2D transformation of the laser points into a regular grid is performed according to equation (2.2), where the column and row in the gridded raster represent respectively its latitude and longitude. Each pixel has a value which represents the elevation.

The rasterisation process often requires interpolation to partly cover for missing data (e.g. filling holes on planar surfaces such as roofs). There are many alternative interpolation techniques such as linear interpolation, nearest neighbour (NN) selection, and inverse distance weighting (IDW) interpolation. According to Arefi (2009), the NN introduces less interpolation error, especially at building edges. However, in low density point clouds, the linear interpolation performs better than the NN.

In the Figure 4.3 example the red cross identifies a pixel which does not have a corresponding value in the point cloud. Indeed, this pixel belongs to a building as can be seen in the orthophoto, but due to reflection errors, there are no points available at or close to the location of the red cross in the point cloud data. Using linear interpolation, the pixel has been assigned an elevation equal to 20.4 m, approximately 2 meters above ground, and coincides with the surrounding points. By assigning this value, it increases the possibility for correctly identifying the pixel as an off-terrain pixel. In contrast, this pixel has been assigned an elevation value equal to 18.56 m using the NN method, which can lead to misclassification error. The linear interpolation technique is utilised during rasterisation of point cloud data for this thesis.

The linear interpolated elevation value Z of a pixel C (i.e. Z_C) located between two known pixels A and B can be computed as follows:

$$Z_C = \frac{\overrightarrow{d_{CB}}}{d} Z_A + \frac{\overrightarrow{d_{CA}}}{d} Z_B, \quad (4.2)$$

where d is the interval distance between A and B; while $\overrightarrow{d_{CB}}$ and $\overrightarrow{d_{CA}}$ are the horizontal distance between C and the known points B and A respectively.

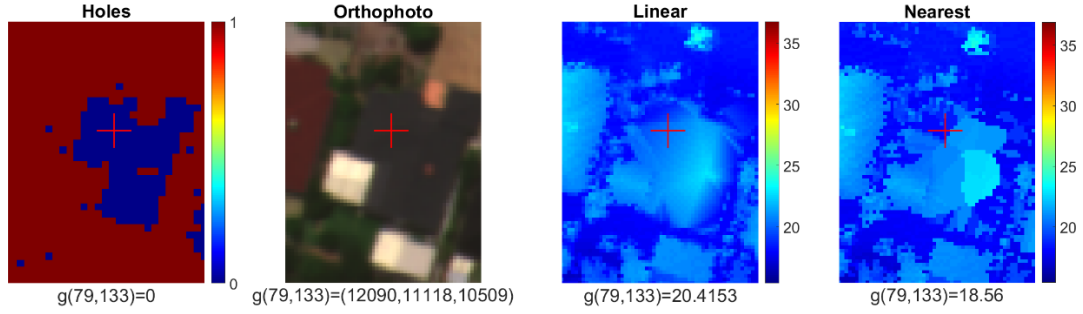


Figure 4.3: Comparison between interpolation methods. Left to right: mask representing missing data/holes, orthophoto, DSM created using linear interpolation, and nearest neighbour interpolation. The red cross indicates missing data, by Linear interpolation method partially covered this missing data (DSM value = 20.4153 m), while by Nearest method, (DSM value = 18.56 m) the same as to the ground.

4.3 DTM and nDSM extraction

This section details the DSM refinement step then presents the proposed DTM extraction algorithm.

4.3.1 DSM refinement

It is well-known that building outlines often appear jagged due to inadequate sampling of the point cloud data. Moreover, a DSM may still have outliers that were not detected and removed. Therefore, a refinement step for the DSM may be necessary, which is then used for creating the nDSM. The degree to which noise affects the building edges depends firstly on the density of the point cloud. In order to mitigate this problem, a median filter (*MF*) is applied. Small elevated objects may disappear in this step, but building outlines will be modelled more precisely as well as improving the DSM quality against outliers (Bulatov et al., 2014). Another advantage of DSM refinement is that the associated sharp edges can improve the accuracy of building geometries. In addition, the process of separating DTM from DSM can be performed more efficiently

because the height difference is the most effective and meaningful parameter in this context. The disadvantage if the filtering is too wide is that close adjacent buildings may merge into a single complex shape. This can lead to an increase percentage of false positives, and the regularisation result can then become less reliable. Therefore, the size of the median filter must be chosen carefully.

Figure 4.4(a) shows an example DSM before median filtering. Refinement results for the described methodology are shown in Figure 4.4(b). Figure 4.4(c) presents the cross section profile taken over a building before (red line) and after (green line) applying the median filter. It can be seen that the noise at a building edge is suppressed when using the refined DSM.

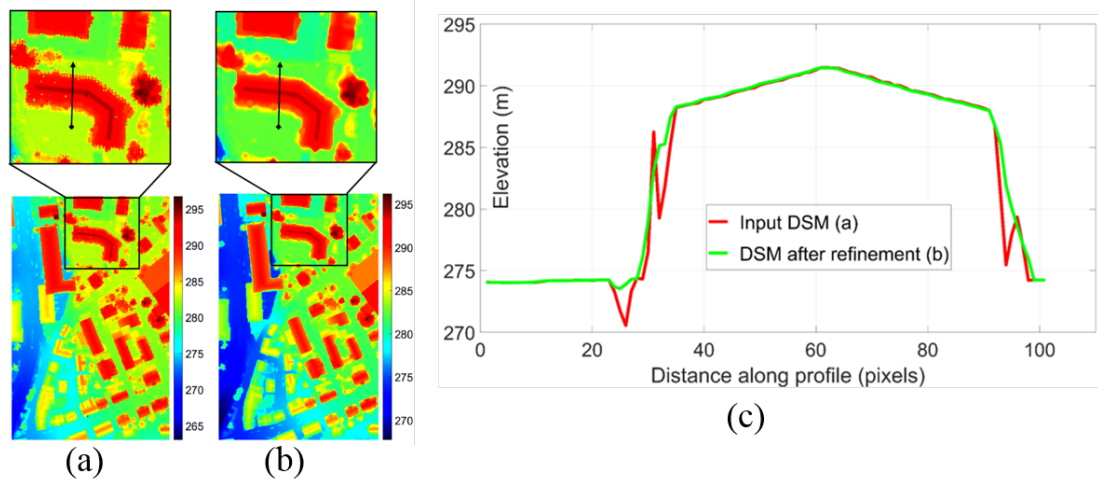


Figure 4.4: DSM refinement: (a) before and (b) after applied median filter. (c) illustrates cross section profiles taken over a building before (red line) and after (green line) applying the median filter.

4.3.2 The proposed DTM extraction algorithm

Understanding the physical properties of the ground surface is the starting point to develop a DTM extraction algorithm. The main geometrical parameters for describing a ground point are the elevation in local area, slope measure, and elevation difference between adjacent points within a certain window size (see section (2.5)). The majority of the existing methods are developed according to these indicators. The proposed DTM extraction algorithm in this thesis is based on the directional scanning

algorithms. The directional scanning algorithm starts with determining the lowest elevation pixel in a scanline, and considers this as a terrain pixel (Perko et al., 2015). The scanline refers to a one directional line where there are a number of pixels located along the line within a certain filtering size for a gridded DSM. For instance, consider the grey pixels in Figure 4.5 as scanlines, the multi-directional and slope dependent method (MSD) (Perko et al., 2015) is presented in Figure 4.5(a), while (b) shows the proposed directional scanlines. The pixel under evaluation (e.g. to be determined if this pixel is a ground pixel or not) is the pixel positioned in the centre of the applied filter. If the height difference between the pixel under examination and the lowest pixel within the mask is more than a pre-defined height threshold, the pixel is categorised as an off-terrain pixel. Otherwise, the slope between the current pixel and the following pixel in the performed scanline direction will be calculated. Then, this pixel is classified as being off-terrain if it exceeds a pre-defined slope threshold.

In general, the drawbacks of existing DTM extraction algorithms are due to either their sensitivity to the slope parameter, or the prior smoothing of DSM (or a combination of both). Regarding the directional scanning methods, other sources of errors can be introduced into the process as follows:

(i) *No ground points in scanlines.*

The determination of ground pixels that are located on the terrain is critical as pixels having the minimal elevation in the scanlines do not necessarily belong to the terrain. This depends on the complexity of the structure in the area of interest, that is, the density of the built-up area, size of the man-made objects, and the dimensions of the applied filter. If incorrect pixels are chosen as minima, misclassification results can occur. Extending the filter size and therefore the length of the scanlines is not an ultimate solution to the problem because it can lead to some raised terrain pixels being wrongly determined as being off-terrain pixels, especially in sloped regions.

(ii) *Selection of incorrect minima in sloped regions.*

Elevation difference between the pixel under examination (centre of the filter) and the minima can be considerable when dealing with sloped regions. In this

case the minima is often placed at the end of the filter due to the sloping terrain. This can lead to classification error because the local structure of the terrain is not considered. Perko et al. (2015) used ground slope fitting by applying a 2D Gaussian smoothing filter to overcome this limitation. However, such a pre-smoothed DSM can lead to inaccuracies because of the level of smoothing applied to the input DSM. Consequently, misclassification may occur.

(iii) *Slope angle as a measure.*

The slope is the most sensitive parameter for all of the existing DTM extraction algorithms (Zhang et al., 2003; Zhang & Whitman, 2005). The problem is clearly demonstrated in very high resolution DSMs because the differences of slope values between the adjacent pixels are considerable without transitioning from terrain to off-terrain, or vice versa. More details can be found in (Mousa et al., 2017).

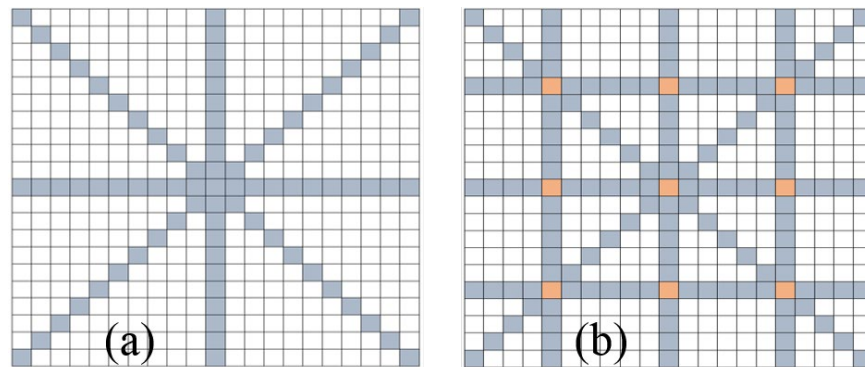


Figure 4.5: Directional scanlines for minimum point selection, (a) MSD approach and (b) the new proposed Network of Ground Points (NGPs) approach.

Based on these drawbacks, a new method of filtering the DSM to establish DTM, described as a Network of Ground Points (NGPs) approach is proposed in this work. Three input parameters are required together with the ground sample distance (GSD) of the input DSM. These parameters are the dimensions of the filter (which should be larger than the maximum building width), a height threshold for judging the validity of ground points, and a parameter for extracting ground mask by analysing pixel heights in an initial nDSM. All thresholds are descriptive and can be set easily. The steps for the proposed method can be summarised as follows:

- (i) Apply new NGPs filter structure (see Figure 4.5(b)) based on the MSD (Perko et al., 2015) (see Figure 4.5(a))- in such a way that well-distributed ground points can be determined.
- (ii) Checking the validity of the selected ground points by setting a height threshold.
- (iii) The accepted terrain points are then stored in their georeferenced position of the original DSM into a 2D grid (Figure 4.6(a)).
- (iv) Generating an initial DTM by filling gaps between the selected terrain points using an interpolation technique (Figure 4.6(b)).
- (v) Generating an initial nDSM.
- (vi) Identify pixels in the initial nDSM having less than a pre-defined threshold as terrain points, as in the equation (4.3). The other pixels are assumed to be off-terrain points and are to be removed (Figure 4.6(c) and (d)).
- (vii) Filling holes associated with removing the off-terrain points to generate the final DTM (Figure 4.6(e)).

$$\text{GroundMask} = (\text{abs}(\text{nDSM}) < \text{threshold}) \quad (4.3)$$

The process of the minima selection of ground pixels in the eight scanlines (grey lines Figure 4.5(b)) is conducted within a pre-defined window size which is moved across the whole DSM raster. This process creates a **Network/seeds of Ground Points**; thus this algorithm is denoted by the name of NGPs algorithm. The intersection points between the crossed scanlines (pixels highlighted in orange in Figure 4.5(b)) are eliminated to avoid scanlines sharing the same pixel as minima. As the designed filter has eight scanlines (Figure 4.5(b)), eight ground points are selected together with their lowest elevation. The lowest point(s) from those eight are optionally omitted to avoid points from regions that are too low (below the ground). Then, height differences among the rest of the points are analysed. A point that has an extremely high elevation difference and exceeds a threshold should be omitted as well. This threshold is introduced in order to overcome the first drawback, that is, where no ground points can be found in a scanline. For instance, if all pixels in a scanline belonging to an off-

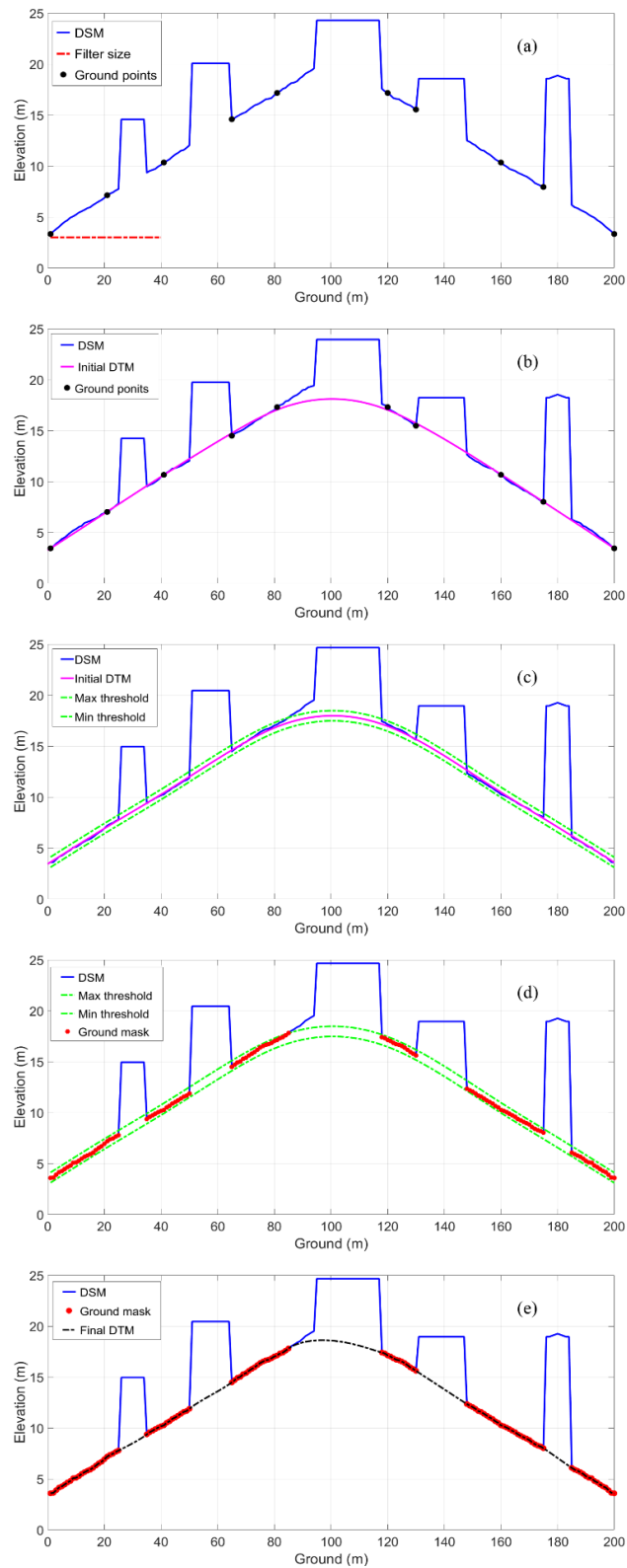


Figure 4.6: Steps for the DTM generation by the NGPs algorithm. Ground points (GPs) selected as minima on an artificial DSM (a), initial DTM (b), and vertical threshold limits (c), ground mask (d), and final DTM (e).

terrain area, then the minimum pixel in this scanline also belongs to an off-terrain area. Thus, it is anticipated that the height difference for this specific pixel in this scanline, compare to the minimum pixel value of all the eight scanlines, is higher than a pre-defined threshold. This off-terrain pixel is then omitted. Figure 4.6 illustrates the steps of the method, using a 40 metre window size for the filter, which is moved by 20 metre intervals. The figure shows an example for one scanline. The filter is moved across the whole DSM raster, at the same time the selected ground points are stored with their georeferenced position of the original DSM into a 2D grid. In Figure 4.6(a), a number of ground points (black dots) are selected from the input DSM (blue line) automatically. Then, an initial DTM is created by filling gaps among the selected ground points by the inward interpolation technique. The pink line in Figure 4.6(c) represents the created initial DTM.

Even though, the quality of the initial DTM is quite acceptable for many applications, two more steps (v and vi) are introduced, as presented in Figure 4.6(c) and (d), in order to obtain a more detailed DTM (see Figure 4.6(e)). In these two steps, an initial nDSM is created. For the initial nDSM, pixels having an elevation within a limit (the green dashed lines) are identified to create a ground mask as in equation (4.3). Then, their corresponding pixels from the input DSM are classified as ground pixels (mask) (red dots in Figure 4.6(d)). The classification decision is made based on the height difference between each pixel in the initial DTM with its corresponding pixel from the input DSM. This means no comparison has been made with a pixel somewhere in the scanline (related to the second drawback), nor with its adjacent pixels considering a slope threshold (related to the third drawback). Therefore, the possible errors resulting from the second and third drawbacks are overcome. The final DTM is depicted by the dashed black line Figure 4.6(e).

After a DTM is created, an nDSM describing buildings and trees can be generated by subtracting the DTM from DSM.

4.4 Building segments extraction

The major objectives of this thesis are to identify building segments as well as their outlines. Thus, the focus is put on eliminating trees only which are the dominant non-building object that interferes with building outline extraction. Comprehensive classification of non-building objects other than trees is outside of the scope of this work.

In order to eliminate trees, the NDVI map is generated using equation (2.1). The NDVI values range from -1 to 1. Experimentally, pixels having NDVI values more than a pre-defined threshold (e.g. 0.15) are likely to belong to vegetation. This NDVI threshold (0.15) may differ from area to area, and is dependent on many factors such as season, density and geographic location. Non-vegetation pixels usually have low NDVI values (e.g. less than 0.15). However, if vegetation is located in regions of shadow the corresponding NDVI values are very low and may be missed. To overcome this limitation, the additional planarity measure is considered (see section 2.4.2).

The planarity measure differentiates between planar areas such as terrain and flat roofs (values are close to 1), and areas with large variations in height. Areas of large height variation can be areas close to walls (values are close to 0, because there is a jump in elevation), trees (values are close to 0 except for the tree crown, which constitute smaller regions of higher than 0 values) and inclined roofs (which, depending on inclination angle, have an approximately constant value between 0 and 1).

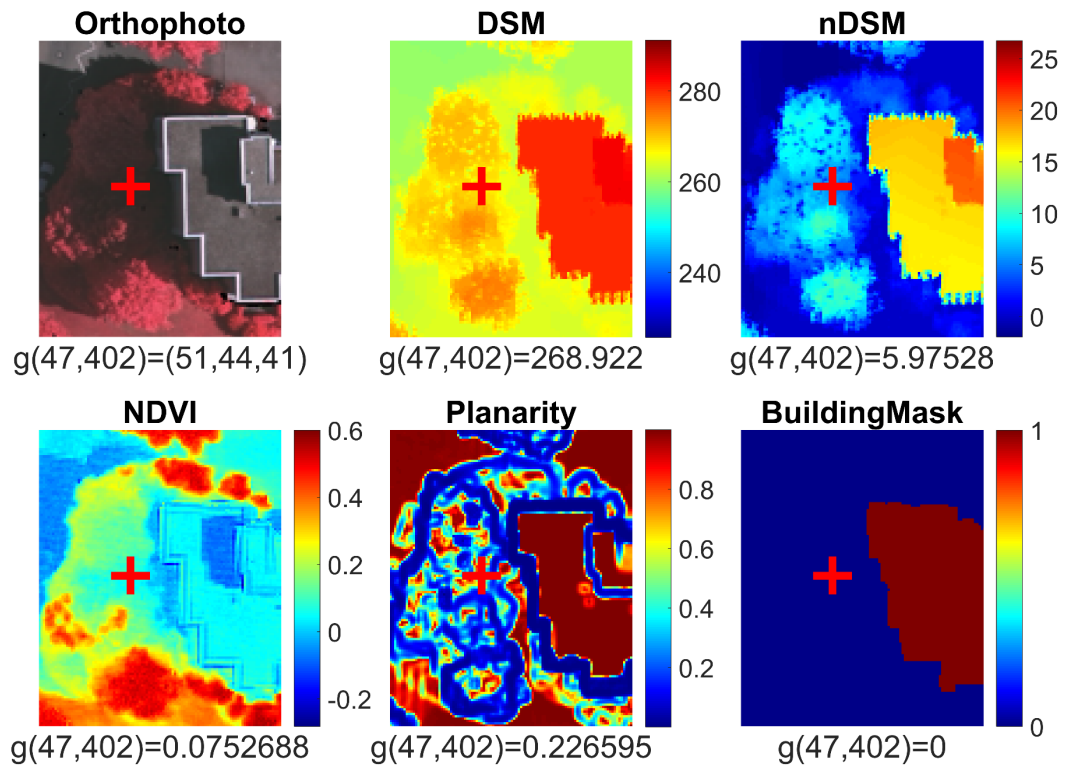


Figure 4.7: Generation of the building mask. Orthophoto, DSM, and nDSM (across top), NDVI, Planarity Map, and Building Mask (across bottom). The red cross belongs to a tree in a shadow area.

The overall decision rule using NDVI and planarity measure is given by:

$$\text{BuildingsMask} = (\text{nDSM} > \text{threshold}) \& ((\text{NDVI} < 0.1) \mid (\text{NDVI} < 0.15 \& \text{planarity} > 0.6)) \quad (4.4)$$

Pixels satisfying equation (4.4) are labelled as building pixels. The result is a binary image separating all building segments (value 1) from the background (value 0). In Figure 4.7, the marked point (denoted by the red cross) belongs to a tree in an area of shadow. Its NDVI value is 0.08, which is too low to be recognised as vegetation, for which the threshold is set to 0.15. However, its planarity measure is 0.22, and therefore lower than the required threshold of 0.6, indicating that this point does not belong to a building. Therefore, the point is correctly eliminated from the nDSM when forming the building mask. It is important to emphasize that the parameters in (4.4) were determined empirically. In addition, small objects (e.g. with an area of less than 10 m²)

are eliminated because firstly, they are likely associated with shadow tree crowns, vehicles, or just noise, and secondly, buildings and building parts with an area less than 10 m² are too small and rarely exist. Finally, morphological filtering is performed to fill small holes in the building mask that may occur.

4.5 Generation of approximate building polygons

The next step includes extraction of approximate building polygons. As an initial step, a list of boundary points for each building should be extracted. Then, the proposed data- and model-driven approaches for the generation of approximate building polygons are applied. Lastly, the proposed labelling algorithm is used to label the boundary points to their corresponding polygon edges. This section explains these steps in detail.

4.5.1 Extraction of boundary points to initialise building polygon

The Moore contour-tracing algorithm (Gonzalez et al., 2004) can be efficiently utilised for the initial tracing of the building outlines. More specifically, the algorithm traces the boundary of a building in a binary image in which pixels having values of one belong to a building, and pixels with a value of zero represent the background. The positions (row and column) of those pixels having both the building object and the background as neighbouring pixels are then identified. This procedure is performed for each single building segment in a scene. In cases where a building segment has a ‘hole’ of zero-valued pixels, it is flagged as a building with an inner polygon. The boundary points for this inner polygon are extracted using the same procedure. Figure 4.8 presents building regions on the left and the extracted boundary points on the right.

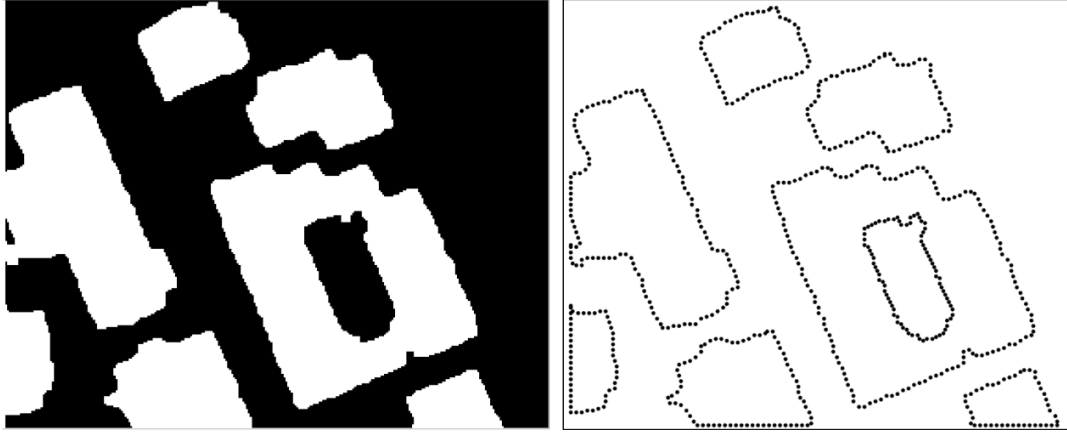


Figure 4.8: Extraction of boundary points. Left to right: building segments and the extracted boundary points.

4.5.2 Data-driven approach for polygon simplification

The goal of the building polygon simplification method is to find the minimum number of vertices through a list of boundary points that can represent a building outline similar to the original building shape. These vertices represent the corners of buildings, where the direction of the boundary points change locally. Connected lines between adjacent vertices are called line segments, representing the shape of a building as close as possible to the real object. In the context of polygons, polylines, and curve simplification, the Euclidean distances between the boundary points and their corresponding line segments is the only criterion considered to solve the simplification problem. The Douglas-Peucker (DP) algorithm considers the distance from the farthest boundary point to its corresponding line segment, which is why it is sensitive to noise. For instance, in a noisy building shape, some boundary points may be located far from their corresponding edges, which may cause the extraction of incorrect vertices depending on the distance/tolerance threshold used. Instead, the building characteristics have to be considered to achieve appropriate level of simplification.

A list of boundary building points is given from the previous steps, and is ordered in any direction (i.e. clockwise or anti-clockwise). Let a building outline be denoted by $P = \{p_1, p_2, \dots, p_I\}$, where $p_i = \{x_i, y_i\}$ is the geometric coordinates of point i located on the border. The goal is to find a simplified building polygon $V =$

$\{v_1, v_2, \dots, v_J\}$, with v being the vertices, and $V \subset P$ and $3 \leq J \leq I$, where J is the number of vertices, V is a closed circular sequence of J vertices ($V_{J+1} = V_1$) (Backes & Bruno, 2013). In this context, the orthogonal distances (E) between each boundary point P_i and its corresponding line segment j are calculated as provided in equation (4.5).

$$Dist = \arg \min_j (dist(P_i, \overrightarrow{V_j V_{j+1}})) \quad (4.5)$$

Figure 4.9 illustrates an example of a list of boundary points which need to be simplified. Starting from an arbitrary point (e.g. indicated by the black triangle V_1 in Figure 4.9(a)), the goal is to find the most probable vertices that define the simplified building outline. Logically, points located at a building corner (e.g. point presented by orange triangle Figure 4.9(b)) should be nominated as vertex V_2 . However, automation of this process is a challenging task. To achieve this the following criteria are considered:

1. *Distance (D)*: V_2 can be defined as the farthest point from V_1 which must achieve three criteria: acceptable *RMSE* (within user defined value), preservation of building area, and the point V_2 should be positioned as close as possible to the corner in order to reduce unnecessary boundary points.
2. *Error (E)*: The *RMSE* of the boundary points between V_1 and V_2 must be acceptable.
3. *Area (A)*: While reducing the unnecessary points between V_1 and V_2 , the area enclosed by the remaining points should be as close to the original area as possible.
4. *Angle (Θ)*: V_2 must not be located on a straight line considering the previous and subsequent points. This criterion is established for identifying building corners (vertices). The priority is given to angles close to 90 degrees.

These four criteria are essential to solve the problem of building outline simplification from a list of given points. To enable automation, these criteria need to be expressed as a mathematical model that may be converted to computer code.

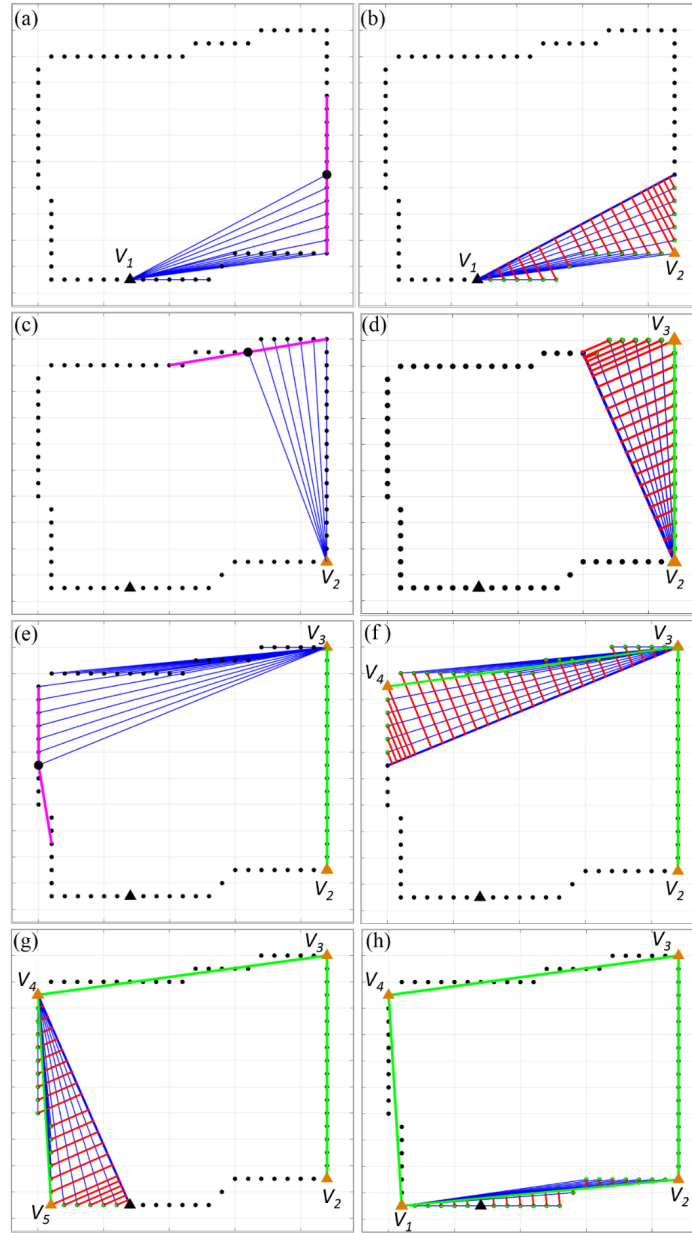


Figure 4.9: Polygonal simplification procedure from given boundary points (black dots). The first starting point (randomly selected) is shown by a black triangle, while the extracted vertices are depicted by orange triangles. The constructed lines, orthogonal distances, and polygon edges are depicted by blue, red, and green lines, respectively. Angles at the points under evaluation are depicted by pink lines. (a) shows constructed straight lines from the start point to the following points. (b) shows the orthogonal distances when the RMSE exceeds a threshold, alongside with the first selected vertex V_2 . In (c), the process is repeated starting from V_2 , and a new selected vertex V_3 is shown in (d). similarly, (e), (f) and (g) shown the later selected vertices V_4 and V_5 . (h) shows the final extracted polygon alongside with a new reorder of the selected vertices.

Based on the introduced building characteristics, the following equation is for evaluating the likelihood of a point being a vertex is proposed:

$$l(i) = A + aD|\sin(\theta)| - bE^2 \quad (4.6)$$

The equation consists of three major parts emphasising the geometrical building criteria such as area preservation and corner detection. The first part presents the calculated area (A) after reducing redundant boundary points.

The second term combines the distance (D) and the angle (θ) between adjacent line segments. D is computed from the starting point to the point under evaluation (last point in the constructed straight line, such as the blue lines in Figure 4.9). θ is the calculated angle between the line segments at the point under consideration. A higher weight for this term is suggested to make sure the distance between two adjacent vertices is maximised within an acceptable error range. A weight of $a = 20$ was determined by empirical testing. This fitting parameter is successful when the evaluated boundary point lies close to a building corner. Otherwise, more weight on the distance may not be preferable, especially if the evaluated point is located on a straight line. For this reason, the weight is multiplied by the absolute sine of the angle (θ) at the point under evaluation. The angle is calculated by using the boundary points preceding and succeeding the current point. The number of these preceding and succeeding points are determined by a threshold called angle detection (AD) to make the two sides required for calculating this angle. Figure 4.9 displays these two sides depicted by two pink lines intersected at the point under evaluation. When the point under evaluation is located at a building corner as shown in Figure 4.9(b), the angle is approximately 90° (or close to 90° in noisy data). In contrast, when the point under evaluation is located on a straight line (e.g. Figure 4.9 (a)), the angle is close to 180° . Since $|\sin(90^\circ)| = |\sin(270^\circ)| = 1$ (maximum value), while $|\sin(180^\circ)| = 0$ (minimum value), the evaluated point collects more weight when it is located at a corner. In contrast, it collects zero weight when it is located on a straight line. Thus, multiplying by the sine of an angle increases the possibility for extracting or preserving building corners/vertices.

The last term (E) represents the MSE of the orthogonal distances (red lines in Figure 4.9) when constructing straight lines (blue lines in Figure 4.9). The orthogonal distances for each boundary point and constructed line are calculated according to equation (4.5). The MSE of the values are then computed, squared, and multiplied by the weight factor b . This factor was set to $b = 2$ based on empirical testing and again can be kept fixed.

The process recursively constructs straight lines from the starting point to the next points as shown in Figure 4.9. At each recursion, a likelihood value is computed according to equation (4.6), and the value is indexed/stored. This process stops when reaching the maximum of the pre-defined RMSE threshold. Then, the boundary point with the highest likelihood value is taken as a vertex. The process is repeated, starting from the last vertex accepted, until returning to the first selected vertex V_2 . Thus, the algorithm is more robust against the effects of the starting point than the DP algorithm. Boundary points that achieved the highest likelihood value at each process iteration will be considered as the most likely vertices.

It is important to mention that in some cases, after reducing redundant boundary points, the calculated area A (the first term in equation (4.6)) is larger than the original area. This is typically occurring when the interior angles of a building are larger than 180° . In this case, the calculated area has to be penalised. For example, if the calculated area ($A_{calculated}$) is more than the original area ($A_{original}$) then term A must be reduced. Firstly, the difference is computed as:

$$\text{Difference} = A_{calculated} - A_{original} \quad (4.7)$$

Then, term A computed as:

$$A = A_{original} - \text{Difference} \quad (4.8)$$

4.5.3 Model-driven approach for outlining rectangular buildings

In this thesis, only the first level of detail for the Recursive Minimum Bounding Rectangle (RMBR) is implemented to estimate whether a building has a rectangular border. This approach is applied to deal with buildings which have high levels of noise

at their border caused by inadequate data, or when occluding vegetation covers parts of building roof. Therefore, the data-driven approach may not always provide the appropriate approximate polygons for such buildings.

The decision whether a building has a rectangular shape is made by thresholding the so-called *similarity ratio* (SR):

$$SR = (A_s / A_m), \quad (4.9)$$

where A_s is the area of the building object (the yellow area surrounded by black boundary points in Figure 4.10(d)), and A_m is the area of the derived MBR as presented by the red box of the same figure. This ratio was considered by Kwak and Habib (2014) to evaluate their building footprint reconstruction result. Figure 4.10 shows an example for a rectangular building represented by its reference yellow polygon in Figure 4.10(a) and DSM data in Figure 4.10(b) showing a very noisy border Figure 4.10(c).

The SR value is always close to one when a building is rectangular, while it becomes smaller with decreasing probability of being a rectangular building. If this value is within the range of a user-defined threshold, a decision of being a rectangle will be made. For a building being estimated to be a rectangular, an approximate polygon with only four vertices depicted with black dots and connected by the green lines (Figure 4.10(e)) is presented whereby all boundary points have been assigned with diverse colours according to their corresponding line segment. The vertices have been selected from the boundary points as they have minimum distance to the corners of the bounding box. Finally, Figure 4.10(f) shows the final building outlines presented as black solid lines associated with up-to-date labelling of the boundary. The advantages of the SR value, is that it can overcome the high level of noise in the boundary points more efficiently. However, the disadvantage is that small details in building outlines may disappear as shown in Figure 4.11.

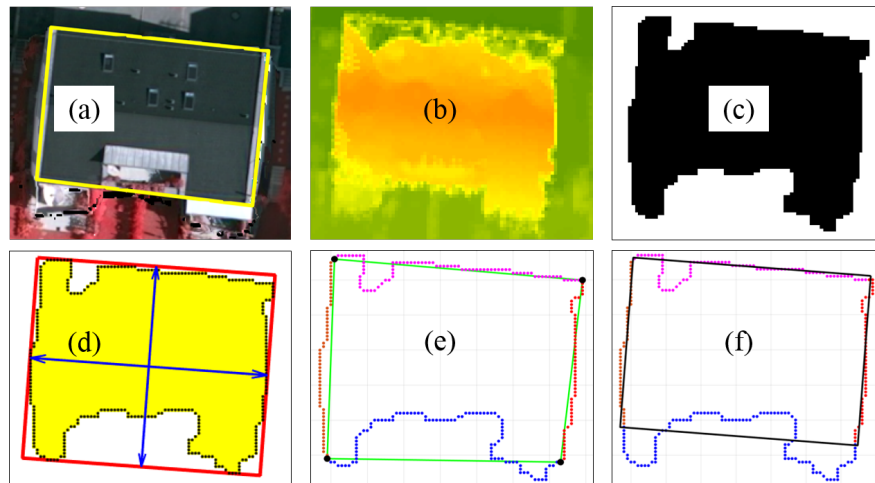


Figure 4.10: Building simplification procedure based on the *SR* showing advantages. (a) showing a building with its reference (yellow polygon). DSM data (b). Building segment (c). MBR (d). Initial and final building outlines are shown in (e) and (f) respectively.

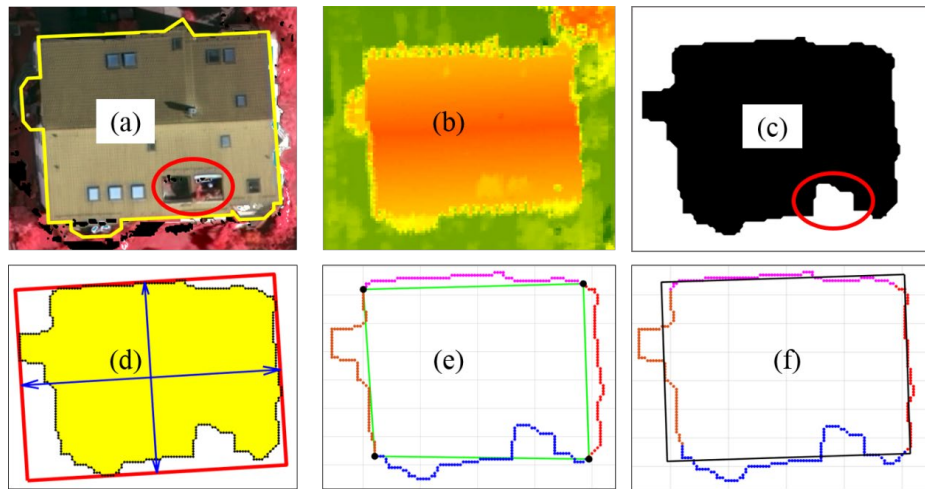


Figure 4.11: Building simplification procedure based on the *SR* showing disadvantages. (a) showing a building with its reference (yellow polygon). DSM data (b). Building segment (c). MBR (d). Initial and final building outlines are shown in (e) and (f) respectively.

4.6 Modelling and adjustment of building polygons

This step aims to perform the best-fit to the boundary points in order to retrieve the regularised building polygon and to enhance the geometric accuracy and regularity of

the building footprints. The Gauss-Markov Model (GMM) and the Gauss-Helmert Model (GHM) are both introduced for modelling rectilinear buildings (Avbelj, 2015).

In order to regularise rectilinear buildings, constraints enforcing rectilinearity have to be implemented. The GMM adjustment deals with observation equations formulating the building outlines. These observation equations can be: 1) orthogonal distances between the boundary points to their corresponding polygon edges; 2) cosine of angles of the building corners; and 3) differences between the estimated and corrected positions of the vertices as proposed by (Avbelj, 2015). The first set of the observations aim to perform the best-fit of the boundary points and their corresponding edges. The second set targeting the creation of right angles. The last set aims to minimise the given corrections to the estimated positions of the vertices. In this model, the orthogonality between the adjacent edges were introduced as pseudo-observations rather than as an extra constraints (Avbelj, 2015). Empirically, it is not guaranteed to obtain exact right angles using GMM. Thus, relaxed polygons are obtained. This is because the model copes with three different problems: best-fit, rectilinearity, and minimising the correction values given to vertices positions. An additional constraint to enforce the rectilinearly was not applied. The rectilinearly of the building polygons output is highly dependent on the quality of approximated positions of the vertices. Thus, the GMM is modified to deal with non-rectilinear buildings since relaxed polygon line segments are obtained.

For modelling of rectilinear buildings, the Gauss-Helmert Model (section 2.6.2) is combined with constraints to enforce rectilinearly. The model has been implemented by Avbelj (2015) but the method of how to implement the constraint in the adjustment was not determined.

In this thesis, the model is solved as explained in the following sections. In order to perform the adjustment, labelling boundary points to their corresponding polygon side must be carried out in advance. Additionally, the positions of the building polygons are modified during the adjustment. Thus, updating the labels of the boundary points is required in order to improve the accuracy of the building outlines.

4.6.1 Algorithm for labelling/assigning boundary points

A list of consecutive boundary points P_i where $i = 1, 2, \dots, I$ boundary points, is given. Also given are the polygon's vertices V (see section 4.5.2), which are the basis for the labelling process. Initially (before any boundary adjustment) V is a subset of P and in the same order clockwise or anticlockwise. However once adjustment begins this is not always the case, meaning that the coordinates of the approximated vertices are not generally from the given boundary points.

An automatic labelling algorithm for updating the assignment of boundary points to their corresponding edges has not been implemented in the literature. To fill this, need a novel algorithm to perform this labelling is introduced to enable improved iterative building outline adjustment. Two ways are possible which are based on either a distance or an angle constraint.

Distance-constraint: one possible approach for labelling the boundary points to their corresponding edges is by considering the orthogonal distances (red lines in Figure 4.12) between boundary points to the polygon edges. A point is assigned to the edge that has the minimum orthogonal distance when compared to the other edges as follows in equation (4.5).

An accurate labelling result can be achieved for rectangular polygons if the approximated vertices are close enough to the actual edges, as can be demonstrated in Figure 4.12(a). However, such an approach fails in some cases such as shown in Figure 4.12(b). The figure shows four boundary points highlighted by the black circle, which are wrongly assigned to the algorithmic extension of the black edge due to their minimum distance to this edge. A similar labelling error is indicated by the blue circle. Thus, a more robust algorithm is needed.

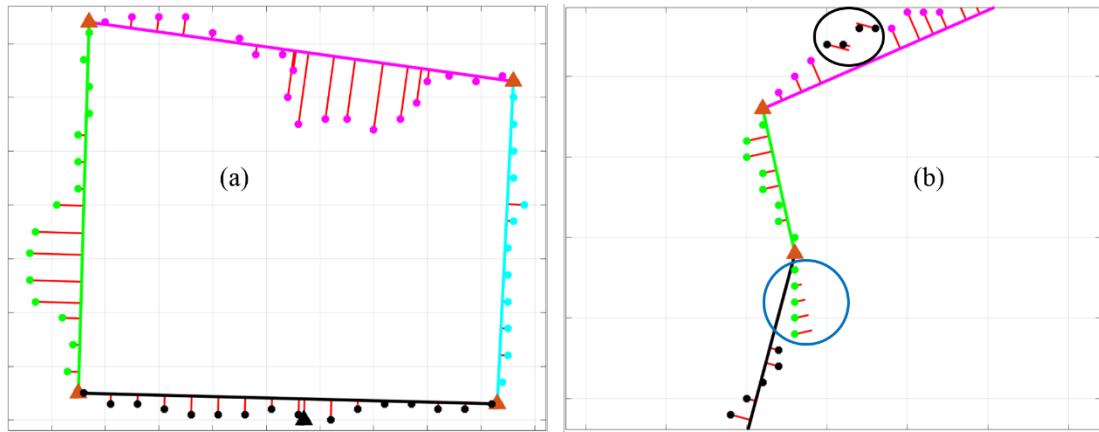


Figure 4.12: *Distance-constraint* algorithm for labelling boundary points to corresponding edges. (a) shows a successful implementation while (b) shows its limitation where the black and blue circles demonstrate mislabelling errors.

Angle-constraint: Alternatively, the assignment approach can be based on the angle measurements between vertices and boundary points. First, the closest vertex to the first boundary point in the list should be estimated. Figure 4.13 shows that V_3 is the closest vertex (orange triangle) to the starting point P_i (black dot). Then, the angle between the line $\overrightarrow{P_i V_3}$ and the previous edge $\overrightarrow{V_3 V_2}$ or ($j = 2$), as indicated by the black line with double-sided arrows in Figure 4.13, is calculated. The angle between the line $\overrightarrow{P_i V_3}$ and the next edge (anti-clock wise order in this example) $\overrightarrow{V_3 V_j}$ or ($j = 3$) is also calculated. A point is labelled according to the edge that yields a smaller angle. If the previous angle (PA) is larger than the next angle (NA), a point is labelled to the previous edge. Sometimes, a boundary point and a vertex are the same, which means the angle is undefined (NaN) or both angles are equal. Thus, the probability for labelling such point to both edges is equal. In the case that the previous angle (PA) is smaller than the next angle (NA), the current point under evaluation is labelled to the current edge j . The next boundary point will be evaluated by comparing the angles around the next vertex ($V = V+1$). The process is performed sequentially until returning to the starting point. When moving from the last vertex ($V = J$, which is equal to the number of polygon vertices) to the next vertex ($J+1$), the algorithm compares angles around the first vertex ($V = j = 1$). Pseudocode for this algorithm is provided in Algorithm 1.

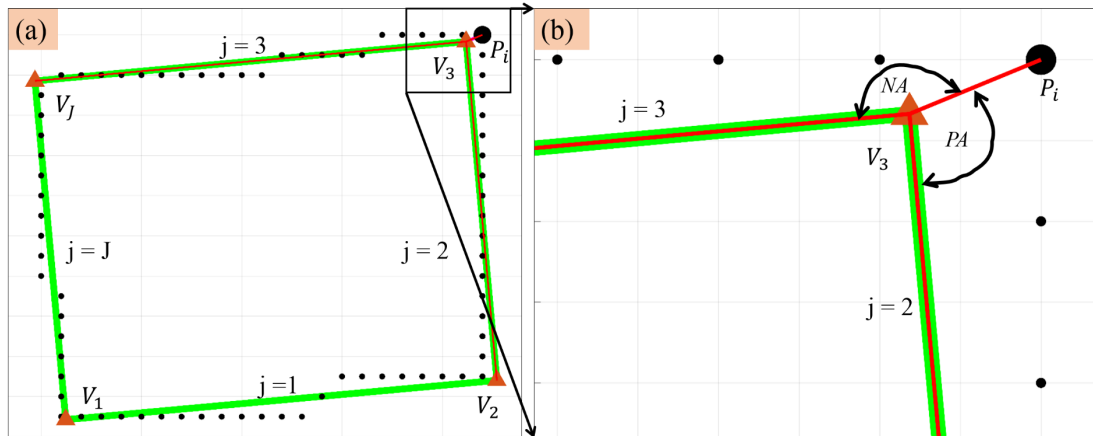


Figure 4.13: *Angle-constraint* algorithm for labelling boundary points to their corresponding edges.

In complex cases, the angle-based method may also have difficulties. This is because the method needs correct estimations of the previous and following angles, which depends on the position of the closest vertex. For example, the closest vertex may belong to an edge from the other side of the polygon as indicated by red circle in Figure 4.14. In this figure, the boundary point P_i should be labelled to the first edge ($j = 1$). However, considering (V_{j-1}) is the closest vertex, the previous and following angles are incorrectly identified. Thus, inaccurate labelling may occur. Therefore, *distance-* and *angle-constraint* methods are combined to provide one concrete solution that is able to solve the labelling problem completely.

Algorithm 1 The labelling / assigning boundary points to their corresponding edges

```

1:  Procedure Label(point  $P_i$ )
2:    for  $i = 1$  to  $I$  do
3:       $j$  = number of the closet vertex to  $P_i$ ,  $V = j$ 
4:      if  $j == 1$ ,  $V_{j-1} = V_J$ , end if
5:      Compute the previous angle (PA) , angle  $P_i V_j V_{j-1}$ ,
6:      Compute the next angle (NA) , angle  $P_i V_j V_{j+1}$ 
7:      if  $PA \leq NA$  &  $j == 1$ 
8:         $L_i = J$  (Label  $P_i$  to the last edge  $J$ )
9:      else if  $PA > NA$  or ( $PA == NaN$  or  $NA == NaN$ )
10:         $L_i = j$  (Label  $P_i$  to the edge  $j$ )
11:      else if  $PA \leq NA$  &  $j > 1$ 
12:         $L_i = j - 1$  (Label  $P_i$  to the previous edge ( $j-1$ ))
13:      end if
14:    return
15:  end procedure

```

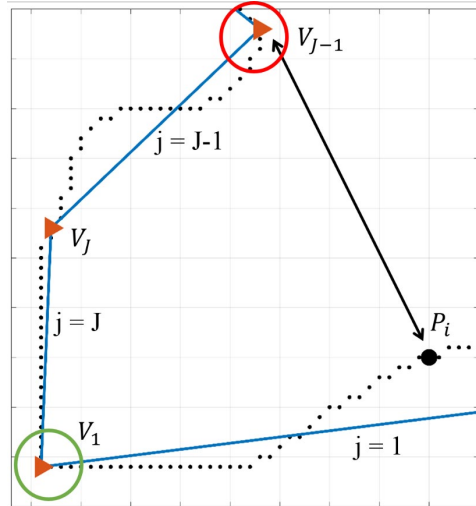


Figure 4.14: Problem of determination of the closest vertex. The figure shows that the closest vertex to the boundary point P_i is (V_{j-1}) . This leads to incorrect identification of the previous and next angles.

Combined distance- and angle-constraint: for each boundary point, the method starts with finding its closest edge by calculating the orthogonal distance as in the equation (4.5). This orthogonal distance must be perpendicular on the edge, not on its extension. Next, the closest vertex to this boundary point is found. If the distance from this boundary point to the closest edge is less than the distance to the closest vertex, this boundary point is labelled as belonging to this closest edge. Otherwise, the *angle-constraint* method is applied in order to make a decision.

However, as mentioned previously, the minimum distance may belong to the extension on the edge, not to the edge itself as shown in Figure 4.12(b). To overcome this limitation, a condition for investigating which point belongs to an edge or to an extension of edge is introduced, as described in Figure 4.15. Firstly, the orthogonal distance is calculated as in the equation and the intersection point E is found (the red square in Figure 4.15). If the distance from T to the furthest vertex in the edge (V_2 in this example) is larger than the length of the edge itself, this boundary point is determined as belonging to the extension. Sometimes, extended edge leading to the false off-set distance is difficult to identify(?). In this case, the *angle-constraint* method is considered.

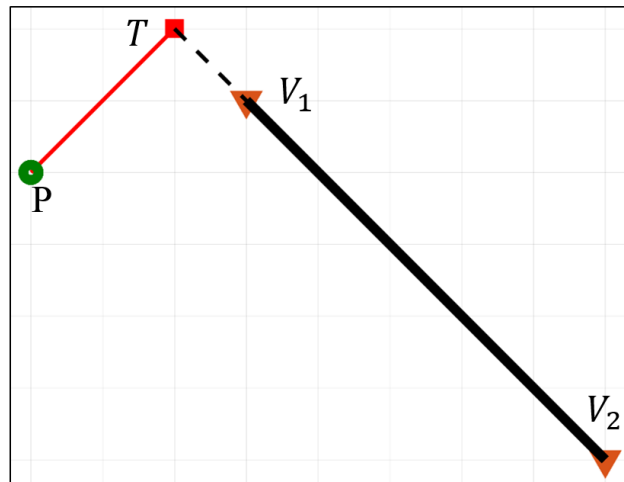


Figure 4.15: Boundary point (P) belonging to an extension of edge ($V_1 V_2$).

Algorithm 2 The labelling / assigning boundary points to their corresponding edges

```

1:   Procedure Label(point  $P_i$ )
2:       for  $i = 1$  to  $I$  do
3:           for  $j = 1$  to  $J$  do
4:                $Dist2edge(j) = \arg (dist(P_i, \overrightarrow{V_j V_{j+1}}))$ 
5:               Find the coordinates of  $T$  ( intersection of  $i$  on the edge  $j$ )
6:               Check if  $X_{T(j)}$  and  $Y_{T(j)}$  located on the edge ( $j$ )
7:           end
8:            $Dist1 = \min(Dist2edge(j)) \ \& \ ((X_{T(j)} \text{ and } Y_{T(j)}) \subset \text{edge } j)$ 
9:            $K = j$  (number of the closest edge )
10:          for  $j = 1$  to  $J$  do
11:               $Dist2Verex(j) = ((X_{p(i)} - X_{V(j)})^2 + (Y_{p(i)} - Y_{V(j)})^2)^{0.5}$ 
12:          end for
13:           $Dist2 = \min ( Dist2Verex(j))$ 
14:           $R = j$  (number of the closest vertex )
15:          If  $Dist1 \leq Dist2$ 
16:               $L_i = k$  (Label  $P_i$  to the closest edge ( $k$ ))
17:          elseif
18:              Compute the previous angle ( $PA$ ) , angle  $P_i V_R V_{R-1}$ ,
19:              Compute the next angle ( $NA$ ) , angle  $P_i V_R V_{R+1}$ ,
20:              if  $PA \leq NA$ 
21:                   $L_i = R-1$  (Label  $P_i$  to the previous edge ( $R-1$ ))
22:              else
23:                   $L_i = R$  (Label  $P_i$  to the previous edge ( $R$ ))
24:              end if
25:          end if
26:      end for
27:      return
28:  end procedure

```

4.6.2 Refinement of non-rectilinear building polygons using GMM

The Gauss-Markov Model (GMM) is used for the adjustment of observation equations. The functional model is assumed to be linearised as described in the equations (2.8) and (2.9). Accordingly, three categories of the observation equations were formulated: squared distances d^2 between each boundary point and its relevant line segment of a polygon represented by F_d as in equation (4.10), observation equations for the interior angles α indicated by F_α , and observation equations for the polygon vertices named F_v . As the coordinates of boundary points $p_i = \{x_i, y_i\}$ are the actual observations, the first set of d^2 are derived observations, while both of the cosines of angles α and polygon vertices are pseudo-observations (Avbelj, 2015).

$$d^2(i, j) = \left(\frac{(Y_j - y_i)(X_{j+1} - X_j) \pm (X_j - x_i)(Y_{j+1} - Y_j)}{\sqrt{(X_{j+1} - X_j)^2 + (Y_{j+1} - Y_j)^2}} \right)^2 \quad (4.10)$$

The denominator of equation (4.10) represents the orthogonal distance between two adjacent vertices (V_j, V_{j+1}) and is positive and greater than zero since $V_j \neq V_{j+1}$. However, if a boundary point i has similar coordinates as a vertex V , the numerator of equation (4.10) becomes zero, and therefore the equation is not differentiable due to

the numerator being an absolute value. Hence, the first set of observation equations F_d is transformed to the squared distance d^2 (Avbelj, 2015).

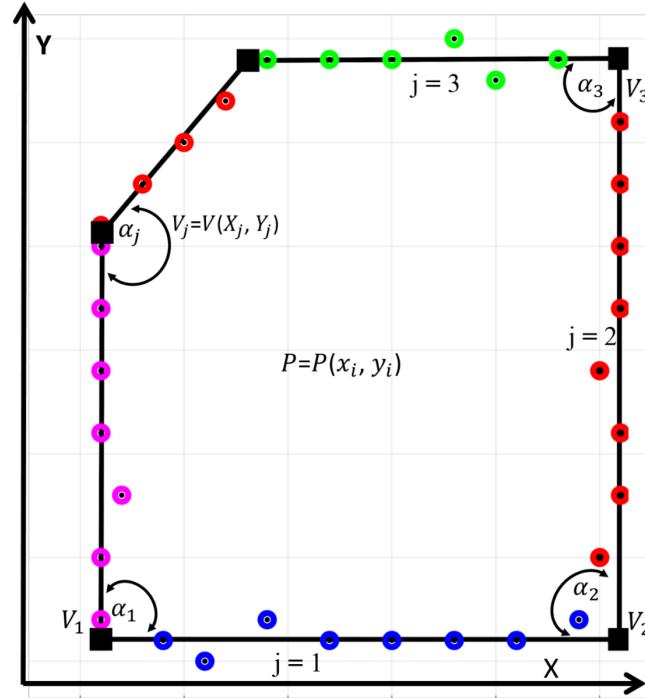


Figure 4.16: Gauss-Markov functional model for adjustment of non-rectilinear building polygons. Boundary points p_i which correspond to the adjusted polygon edges (black polygon) are specified (dots in different colours). Coordinates of vertices V_j (black squares) are the unknown parameters u . The vector of observations y comprises the squared distances between boundary points and polygon edges $d^2(i, j)$ (derived observations), cosines of angles α_j and vertices of the polygons V_j .

The second set of the observation equations F_α represent the cosines of the interior angles, which are either 90° or 270° . Since $\cos(\frac{\pi}{2}(2k + 1))$ is always equal to zero for $k \in \mathbb{Z}$, the cosine of an interior polygon angle can be computed as follow:

$$F_{\alpha(j)} = \frac{(X_j - X_{j-1})(X_{j+1} - X_j) + (Y_j - Y_{j-1})(Y_{j+1} - Y_j)}{\sqrt{(X_j - X_{j-1})^2 + (Y_j - Y_{j-1})^2} \sqrt{(X_{j+1} - X_j)^2 + (Y_{j+1} - Y_j)^2}} \quad (4.11)$$

The polygon vertices coordinates (X_j, Y_j) are the last set of observation equations which are formulated as follow:

$$F_{V_{X(j)}} = X_j - X_j^o \quad (4.12)$$

$$F_{V_{Y(j)}} = Y_j - Y_j^o \quad (4.13)$$

Since the polygon vertices coordinates (X_j, Y_j) are unknown, this part of equations could be eliminated from the functional model. But, the newly updated location of the vertices in the adjustment should be preserved near to their estimated positions, and their accuracy can be controlled via weightings (Avbelj, 2015).

It can be seen that the functional model F is non-linear. Therefore, the partial derivatives of F with respect to the unknown parameters u should be taken at the approximated values u^o . The unknown parameters u are the coordinates of the polygon's vertices. The linearised functional model can be written as:

$$A_{(I+3J),u} \delta_{u,1} + w_{(I+3J),1} = v_{(I+3J),1} \quad (4.14)$$

where A is the design or coefficient matrix $A = \frac{\partial F}{\partial u} |_{u^o}$; δ is a vector of corrections to the unknown parameters u which has the dimension double the number of vertices (corrections to X and Y coordinates of each vertex); and w is the misclosure vector. Moreover, v is a vector of residuals which has zero expectation (residuals equal to zero in perfect solution).

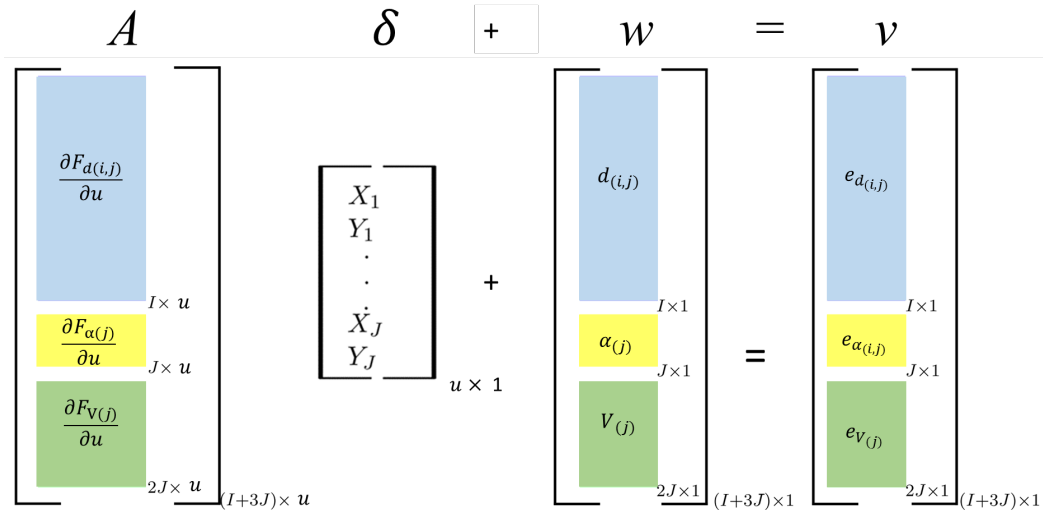


Figure 4.17: Schematic matrix illustration of the GMM equation (4.14) for rectilinear building polygons approximation (modified from (Avbelj, 2015)). Three categories of observation equations were formulated, for squared distances between the boundary points and the building polygon (light blue), for cosines of angles (yellow), and for polygon vertices (green).

The least squares adjustment tries to minimise the square sum of the residuals from the three different set of observation equations F_d , F_α , and F_v . The first set aims to minimise the orthogonal distances between the boundary points and their corresponding polygon edge. The second set F_α aims to minimise the cosines of the interior angles in such a way that the obtained angles are close to 90° or 270° . The last set F_v minimises the correction values of the polygon vertices in such a way that the adjusted vertices location remain close to their initial locations.

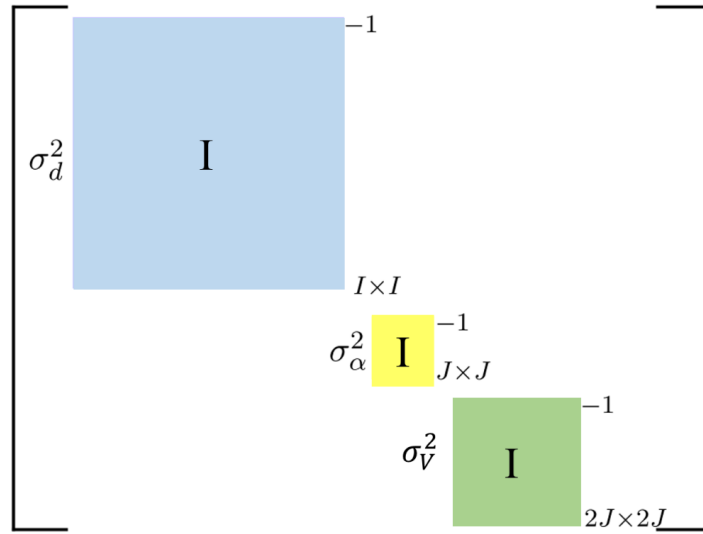


Figure 4.18: Covariance matrix Gauss-Markov Model.

Since the building polygon is non-rectilinear, the goal is to best-fit the boundary points with their corresponding edges. In addition, if right angles exist, they should be optimised to be as close as possible to 90° . These requirements could be achieved through the weights given to the implemented sets of the observation equations. For the first set (F_d), the orthogonal distance (d) of each boundary point (i) to its corresponding line segment (j) is assumed to be the standard deviation ($\sigma_{d(i,j)}$) and is calculated from equation (4.5). In other words, the variance for each boundary point P is calculated as $\sigma_d^2 = 1/d_{(i,j)}^2$. The second set of the implemented observation equations (F_α) involves the cosine of the polygon angles. The variance of each angle in the polygon is calculated as $\sigma_\alpha^2 = 1/\cos_{(\alpha)}^2$. The last set (F_v) contains the spatial difference between the estimated and updated coordinates of building polygons (equations (4.12) and (4.13)). This particular set was designed to keep the adjusted

vertices coordinates close to the estimated values (Avbelj, 2015). Giving high weight to this set (F_V) would impede their further enhancement towards the actual positions. Thus, the identity matrix is used to weight this particular set of observation equations so that the variance is $\sigma_V^2 = 1$. The covariance matrix Σ has $(I + 3J) \times (I + 3J)$ dimensions and is designed as shown in Figure 4.18.

4.6.3 Refinement of rectilinear building polygons using GHM

For modelling rectilinear buildings, the orthogonality and/or parallelism assumptions of the building polygon must be implemented. In a mathematical sense, if a line segment j has a normal vector $n_j = (n_x, n_y)$, then the next line segment must have a perpendicular normal vector $n_{j+1} = (n_y, -n_x)$. All parallel polygon edges should have the same normal vector as depicted by the red arrows in (Figure 4.19).

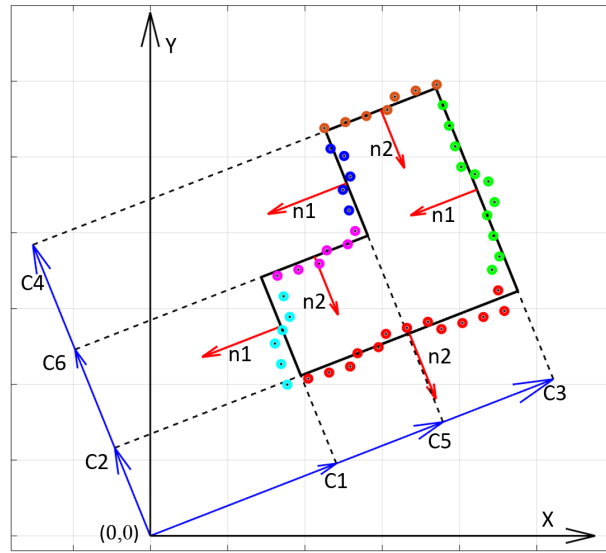


Figure 4.19: The orthogonality functional model: Boundary points (dots) coloured differently according to their corresponding edges. Red arrows indicate the direction of normal vectors (not to scale). The blue arrows are the vector distances C computed from the origin in the local building outline coordinate system. The black solid polygon is the final rectilinear polygon after the adjustment.

The observation equations for a boundary point i which is expressed by its Cartesian coordinates (x_i, y_i) located on its corresponding line segment are given as follows:

$$F_{(j)} = n_x x_i + n_y y_i - C_j = 0 \quad (4.15)$$

$$F_{(j+1)} = n_y x_i - n_x y_i - C_{j+1} = 0, \quad (4.16)$$

where C_j and C_{j+1} are the distances from the centre of the coordinate system perpendicular to the extended line segment j , as shown with blue arrows in Figure 4.19. The unknown parameters (u) are n_x, n_y , and $C_{1,...,J}$, where J is the number of vertices or polygon edges. The normal n_x and n_y must have a length equal to one and satisfy the following constrain:.

$$g = n_x^2 + n_y^2 - 1 = 0 \quad (4.17)$$

Because the unknowns and observations for a point in the line segment observation equation cannot be separated, and each constraint contains more than one observation, the mixed model or GH model adjustment has to be applied (Skaloud & Lichti, 2006). The model is linearised for the approximated values u^o of the unknowns (u) and the observations (l), as in equation (2.12), and written in matrices form as:

$$A_{I,u} \delta_{u,1} + B_{I,2l} v_{2l,1} = w_{I,1} \quad (4.18)$$

where $A = \frac{\partial F}{\partial u}$ is the design matrix of partial derivatives of the observation equations with respect to the unknowns (u) at the approximated values u^o ; $\delta = (\hat{u} - u^o)$ is the correction vector to the approximated values; $B = \frac{\partial F}{\partial l}$ is the design matrix of partial derivatives of observation equations with respect to the coordinates x_i and y_i ; $v = (\hat{l} - l)$ is the vector of residuals calculated as the difference of the adjusted observation vector \hat{l} and the observation vector l ; $w = F(\hat{u}, l)$ is the misclosure vector, e.g. the equations (4.15 and (4.16) are evaluated at the current estimates of the unknown parameters and observations. The matrices A and B are designed as follows:

$$A_{(I \times (2+J))} = \left| \begin{array}{cccccc} x_1 & y_1 & -1 & 0 & \dots & 0 \\ & & \dots & \dots & \dots & \dots \\ & & -1 & 0 & \dots & 0 \\ \vdots & \vdots & 0 & -1 & \dots & 0 \\ & & \dots & \dots & \dots & \dots \\ & & 0 & -1 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ x_{I-1} & y_{I-1} & 0 & 0 & \dots & -1 \\ x_I & y_I & 0 & 0 & \dots & -1 \end{array} \right| \begin{array}{l} j=1 \\ \\ \\ j=2 \\ \\ \\ j=J \end{array} \quad (4.19)$$

$$B_{(I \times 2I)} = \left| \begin{array}{ccccc} n_x & 0 & 0 & \dots & 0 \\ 0 & n_y & 0 & \dots & 0 \\ 0 & 0 & \ddots & \dots & 0 \\ \vdots & \vdots & 0 & n_y & 0 \\ 0 & 0 & 0 & 0 & -n_x \end{array} \right| \quad (4.20)$$

The conditional equation (4.17) is linearised as shown below, and applied as weighted unknown constraint:

$$G_{1,u} \delta_{u,1} + w_{c,1,1} = v_{c,1,1}, \quad (4.21)$$

where $G = \frac{\partial g}{\partial u}$ is the matrix of partial derivatives of the conditional equation (4.17) with respect to the unknowns; $w_c = g(n_x^o, n_y^o)$ is the misclosure vector of the conditional equation (4.17) for the approximated values of n_x and n_y ; v_c is the residual of the constraint. Matrix G is designed as follows:

$$G_{(1 \times (2+J))} = \left[\frac{\partial G}{\partial n_x} \quad \frac{\partial G}{\partial n_y} \quad 0 \quad \dots \quad 0 \right] \quad (4.22)$$

Similar to the weight given to the first set of observation equations F_d in the GMM, the variance of a boundary point ($\sigma_{(i)}^2$) is assumed to be the inverse of the squared orthogonal distance between this point i and its corresponding edge j , and computed

as $1/d_{(i,j)}^2$, where $d_{(i,j)}$ is calculated from equation 2.9. Accordingly, the covariance matrix C for the observations which has dimensions of $(2I) \times (2I)$ is designed as in the equation below.

$$C_{(2I \times 2I)} = \begin{bmatrix} d_{(1,j)}^2 & & & & \\ & d_{(1,j)}^2 & & & \\ & & d_{(2,j)}^2 & & \\ & & & d_{(2,j)}^2 & \\ & & & & \ddots \\ & & & & & \ddots \\ & & & & & & \ddots \\ & & & & & & & d_{(I,j)}^2 \\ & & & & & & & & d_{(I,j)}^2 \end{bmatrix} \begin{matrix} -1 \\ \\ \\ \\ \\ \\ \\ -1 \end{matrix} \quad (4.23)$$

The weight matrix P_c for the constraint must be over-weighted (e.g. given a high weight $P_c = I^2$) which has (1×1) dimensions. Otherwise, for n_x and n_y , an infinite number of solutions are obtained because the scale of the vector distances C cannot be determined. Finally, the adjustment solution is formulated according to the least squares procedure considering the combined model as follows:

$$\delta = -[A^T (BP^{-1}B^T)^{-1}A + G^T P_c G] [A^T (BP^{-1}B^T)^{-1}w + G^T P_c w_c] \quad (4.24)$$

4.6.4 Initial solution and stopping criterion for the adjustment

For a non-linear least squares adjustment, an initial solution of the unknown parameters is required. For the GMM, the unknowns are the vertices coordinates of the building polygons. Those coordinates are already given (see section 4.5.1). During the adjustment, those coordinates are given corrections, and consequently, their positions are iteratively updated. Therefore, the labelling of the boundary points requires updating too.

Compared to the GMM, in the GHM the estimation of initial solution of the unknowns is more complex. The unknown parameters are the normal components n_x and n_y ,

and the vector distances C from the origin to the extension of each of the input polygon line segments as shown in the Figure 4.19. Firstly, the input boundary points have to be labelled in accordance to the polygon line segments/edges as described in section 4.6.1. Then, the longest segment is identified. This longest line is fitted with respect to its corresponding boundary points using line-fitting least squares procedure. Then, the normal values n_x and n_y are derived from the fitted line. After that, the vector distances C are estimated by applying the equations (4.15) and (4.16). In these two equations, the coordinates (x_i, y_i) must be given. In order to estimate the vector distance C_j , the mean of the boundary points belonging to this segment j was taken to find the coordinates (x_i, y_i) . In some cases, the longest line segment may not represent the actual dominant building direction, which may lead to fail the adjustment (diverging solution). Thus, the process of estimating the initial values is repeated considering the next longest segment (until the four longest segments are examined if needed).

In both adjustment procedure (GHM and GMM), the adjustment is terminated when the correction values become insignificant (i.e. less than 0.5 pixel). The root mean square error of all boundary points to their adjusted polygon edges is then calculated. A decision to accept the adjusted polygon is made based on a user-defined threshold. The GHM provides a regular shape by enforcing the rectilinearity conditions. Thus, a large threshold is required if the proposed model-driven approach is used (i.e. three meters).

4.7 RMSE as a measure for accuracy assessment of building polygons

It is very important to assess the obtained polygons after adjustment by comparing them to ground truth data, or to a result from a different method or to the approximated polygons. Two essential factors have to be considered to perform a reliable evaluation procedure. Firstly, a suitable ground truth dataset has to be available. For evaluation process, this dataset should have a variety of different sizes, shapes, rectilinearly, and varying complexity to achieve a reasonable assessment. Secondly a suitable measure of consistency of the outline with respect to human perception must be established.

Several measures have been utilised for assessing the quality of the extracted building polygons similarity. For instance, the well-known RMSE, *Hausdorff distance* (Bulatov, 2011; Olson & Huttenlocher, 1997), and Polygon and Line Segments (*PoLiS*) (Avbelj, Müller, & Bamler, 2015).

In this thesis, the RMSE parameter is utilised due to being extensively used as a standard measure in topographic mappings. Shortcomings of RMSE and how to avoid them is discussed first. Then, a new procedure for assessment is proposed to avoid any subjective evaluation process in the interpretation of the RMSE as a standard metric.

Assume that a reference building polygon model R which contains a number of vertices (e.g. $j = 1, \dots, J$) and an extracted polygon indicated by E which contains a number of vertices (e.g. $k = 1, \dots, K$) are given. The ultimate goal is to assess how close the extracted polygon is to its reference model. The number of vertices in both models are not necessarily the same. Figure 4.20 shows a simulated example measured in pixels. The figure illustrates an extracted building with five vertices depicted by the green polygon, while its reference model (red polygon) has only four vertices.

On the one hand, the RMSE has been calculated between the closest vertices in both models. Figure 4.20(a) shows 3.07 and 1.32 RMSE calculated from the extracted vertices to their closest vertices in the reference and vice versa respectively. We denote this as Vertex to Vertex evaluation (V2V). Considering the huge gap between the values e.g. (3.07 to 1.32), a question is raised about which one is more reliable to reflect the dissimilarity. In fact, this huge gap was caused by the redundant vertex indicated by the blue arrow in Figure 4.20(a). Skipping such a redundant vertex would provide an equivalent RMSE (1.32), but this option should be avoided for two reasons. Firstly, skipping undesirable vertices will lead to a subjective evaluation process. The procedure of vertex to vertex calculation has been considered in several existing studies, such as (Höhle, 2017). The second reason is that redundant vertices should be penalised in order to evaluate the applied algorithms and their results in reliable manner. On the other hand, when considering calculation of the RMSE from vertices to their corresponding edges. This option is not suitable due to some vertices actually

belonging to an extension of an edge and not the edge itself. For example, the two top extracted vertices (Figure 4.20(a) and (b)). Therefore, both strategies, Vertex to Vertex and Vertex to Edge, are not suitable tools for measuring the dissimilarity or correspondence between building extracted and reference polygons.

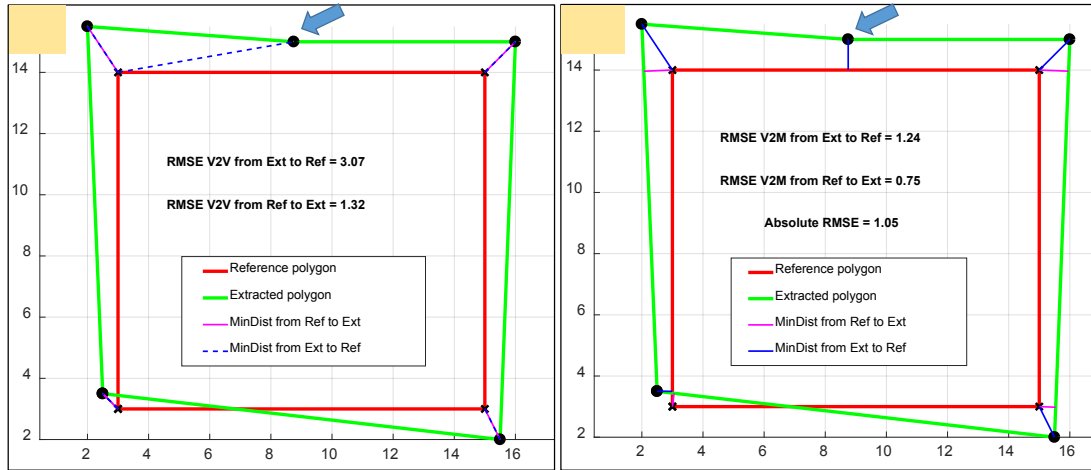


Figure 4.20: Calculation of RMSE. Red polygon represents a building reference while an extracted building depicted by the green polygon. (a) shows a RMSE calculated from minimum distances measured from each vertex in the reference to the closest vertex in the extracted and vice versa. (b) considers minimum distances from each vertex to the closest edge or vertex.

Therefore, it is proposed to calculate the RMSE utilising a combination of the two aforementioned models. For instance, if a vertex belongs to an extension, the distance to the closest vertex is considered. Otherwise, the minimum distance to the closest polygon edge should be applied. This should be performed between the extracted model and the reference model (and vice versa). This is denoted as Vertex to Model (V2M) evaluation. To do this each vertex in the reference polygon should be labelled to its corresponding edge in advance. Similarly, each vertex in the extracted polygon must be also labelled to its corresponding edge. Otherwise, the minimum distances to the closest edge may be wrongly measured. This labelling has been done in an earlier stage (see section 4.6.1 for more details). Figure 4.20(b) illustrates how those distances (blue and magenta) are considered, which result in RMSE values equal to 1.24 and 0.75 from the extracted model to the reference and vice versa respectively, denoted as

vertex to model (V2M). Additionally, and to avoid any subjective evaluation process by considering one result (e.g. 0.75) rather than the other (1.24), the absolute RMSE between both models is calculated as follows:

$$RMSE(R, E) = \left(\frac{1}{J+K} \left(\sum_1^J \left(\min(\text{dst}(\overrightarrow{E}, \overrightarrow{R})) \right)^2 + \sum_1^K \left(\min(\text{dst}(\overrightarrow{R}, \overrightarrow{E})) \right)^2 \right) \right)^{0.5} \quad (4.25)$$

where, J and K represent number of vertices in the extracted and reference polygons respectively. All observed distances from each vertex in E to its closest edge or vertex in R alongside with distances from each vertex in R to its closest edge or vertex in E are squared, accumulated, and the square root taken.

5 Chapter five Evaluation

In this chapter, the details about the datasets used in the evaluation process are provided. Based on these datasets, the proposed DTM extraction algorithm is evaluated and compared with to another state-of-the-art method. Further, the fusion method for the building segment extraction, as proposed in section 4.4 is examined using the evaluation datasets. An assessment and sensitivity analysis of the proposed data-driven method for the generation of approximate building polygons is then presented. Finally, the regularisation results, including model assumptions and the adjustment (GMM and GHM) are quantitatively evaluated utilising the new procedure (as described in section 4.7) for calculating the RMSE.

5.1 Study areas

5.1.1 Vaihingen datasets

The dataset that is used the evaluation is of Vaihingen (Cramer, 2010). Specifically the Areas 1, 2, and 3, as illustrated in Figure 5.1 are used. It is an openly available dataset and commonly used for benchmarking. The sites contain 107 buildings with varying sizes and complexity. The three specific Areas of the Vaihingen dataset were chosen because they contain and represent different building characteristics within each Area, which makes them challenging for accurate building regularisation. In addition, the non-flat nature of the terrain, the presence of high raised buildings, and the inclusion of larger objects located on top of some of those buildings make the process of separating the DTM from the DSM challenging. The specifications of the dataset are:

- LiDAR DSM with 25 cm GSD derived from point clouds data captured by a Leica ALS50 system with 6 points/m² density average.
- Airborne images (8 cm GSD) acquired by the Intergraph/ZI DMC platform with a 12 cm focal length (Cramer, 2010). The colour information consists of three bands: near infrared (NIR), red (R), and Green (G). The orthophoto image is also provided with the dataset.

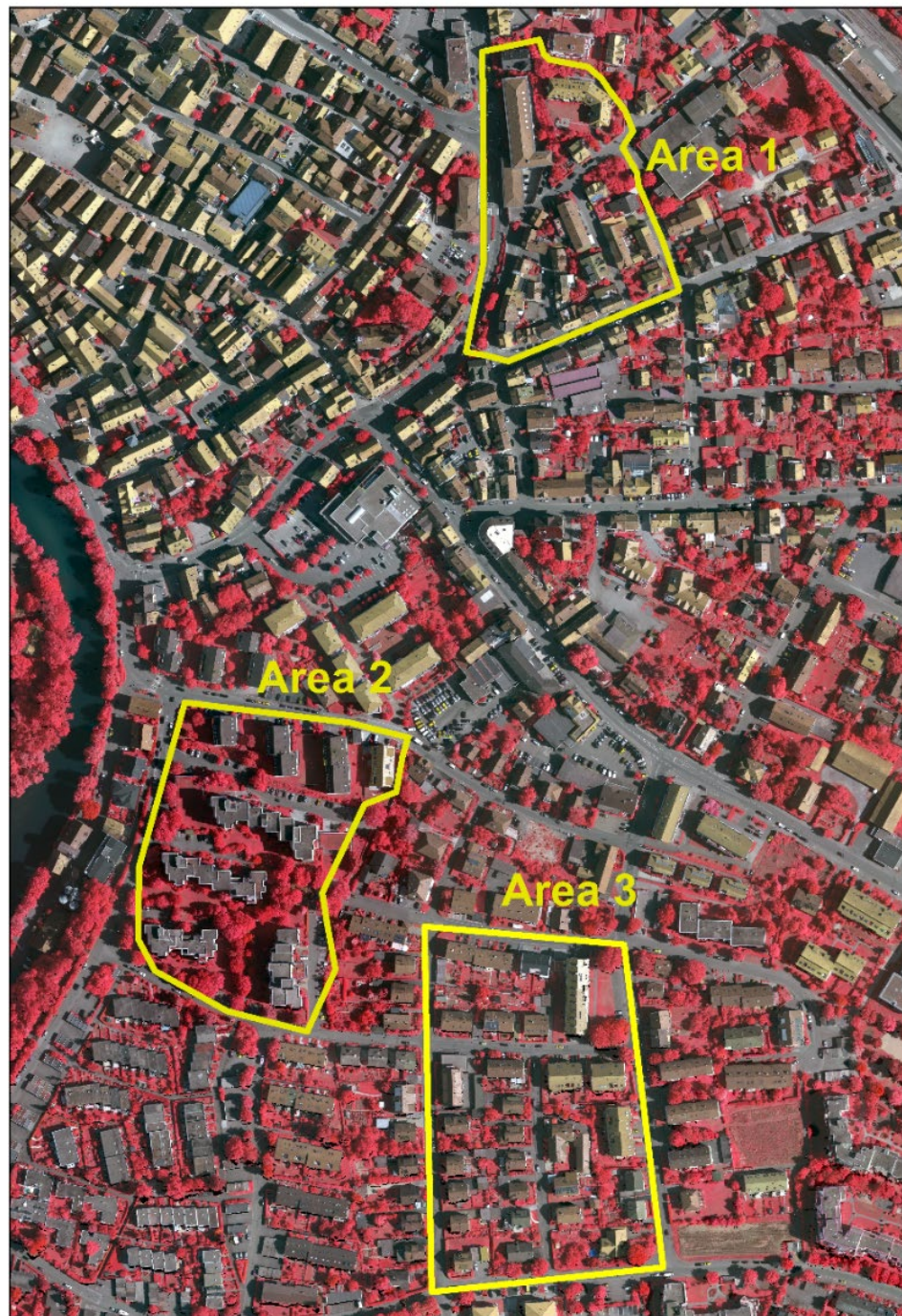


Figure 5.1: Vaihingen study area including the three Areas 1, 2, and 3 (Cramer, 2010).

5.1.2 Melville datasets

While the Vahingen datasets are of high complexity, the dimension of the test areas are relatively small. To consider a large dataset, the City of Melville dataset is also utilised in this thesis for evaluation. The city is located to the south of Perth in Western Australia. The majority are residential buildings of one story and a height of approximately three metres. The building outlines are generally complex. Topographically, the terrain of the city area is hilly, with the elevation increasing from zero to nearly 60 metres with respect to the Australian Height Datum (AHD). An area with approximately 3.6 km² was selected, as shown in Figure 5.2 (which includes nearly 2,200 buildings). Two types of collected data have been provided:

- ALS data: The point cloud data were acquired using a Riegl VZ- 1000 3D laser scanner (Riegl, 2017). This data was captured at an average height of 457 metres height, which yielded approximately 2 - 4 points / m² spatial density and an accuracy of 0.1 meter.
- Multispectral data: the multispectral imagery were acquired by an Unmanned Aerial Vehicle (UAV) platform via the MicaSense RedEdge camera (MicaSense, 2017). The sensor has a global shutter and a 47.2° field of view (FOV). It has five spectral bands: Blue (B), Green (G), Red (R), RedEdge (RE) and Near-infrared (NIR). 0.41 m GSD was achieved during the acquisition. The orthophoto was resampled to 0.5 metres, so as to have the same resolution as the derived DSM.



Figure 5.2: Melville study area.

The ALS data is first cleaned from outliers according to the procedure presented in section 4.2.1, and rasterised according to equation (2.2). This final rasterised DSM is shown in Figure 5.3.

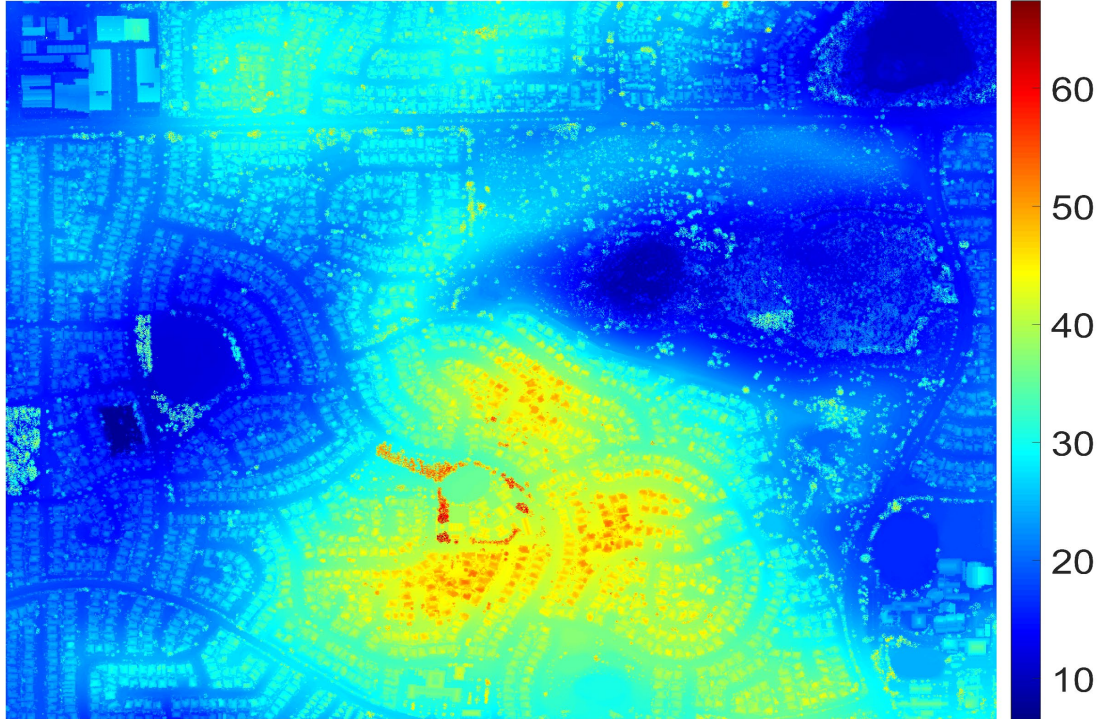


Figure 5.3: The created DSM for Melville dataset. The units of the elevation colour-scale are in metres.

5.2 Evaluation of the proposed DTM extraction algorithm

The DTMs for the Vaihingen and the Melville datasets are created using the proposed Network of Ground Points (NGP) method outlined in section 4.3.2. Ground truth for terrain is not available for either of the datasets. Instead, a reference classification map (layers including buildings, trees, grass, bare-ground, and cars) is provided. Thus the NGP method is applied to the photogrammetric DSM, while a DTM generated from the LiDAR data is adopted as the reference. For the Melville dataset, a photogrammetric DSM is not available, therefore, for the rest of this section evaluation of the DTM extraction algorithm, the Vaihingen dataset is used only. More specifically, Areas 1 and 2 of the Vaihingen dataset are selected for this task.

The completeness and correctness are also calculated for extracted bare-ground compared to the reference bare-ground layer provided. Comparison is made with one of the most promising state-of-the-art DTM extraction methods, the Multi-directional and Slope Dependent (MSD) method (Perko et al., 2015) to evaluate the efficiency of the proposed method. It is true that the MSD method was developed to extract DTM from satellite-based DSM, but the method is effective and used by many researchers e.g. (Misra, Avtar, & Takeuchi, 2018).

The reference DTM is created from LiDAR data using LAStools software, (version 141017, academic), obtained from <http://rapidlasso.com/LAStools>. The completeness (C_m), correctness (C_r), and quality (Q_l) can be calculated as follow (Rottensteiner et al., 2014):

$$C_m = \frac{TP}{TP + FN} \quad (5.1)$$

$$C_r = \frac{TP}{TP + FP} \quad (5.2)$$

$$Q_l = \frac{TP}{TP + FP + FN} \quad (5.3)$$

where TP (true positive) represents number of ground pixel identified correctly; FN (false negative) is number of missed ground pixels; FP is the false positive indicates number of non-ground pixels classified incorrectly as ground pixels.

5.2.1 Parameter settings for DTM extraction

The parameters used for the state-of-the-art MSD approach are provided in Table 5.1, and for the novel NGPs approach in Table 5.2. The parameter values for the filtering window size are fixed to 53 m for both approaches and both datasets (Areas 1 and 2). Determining the filtering window size is dependent on the dimensions of the existing buildings in the scene. For the MSD approach, the height threshold is based on the absolute height difference between the ground and buildings. The MSD method is based on a comparison of height difference between a point under evaluation and the point have the lowest elevation in a scanline. This lowest point may belong to regions

that are too low, which lead to misclassification. Thus, the height threshold parameter is set to three meters. The slope parameter is based on the elevation difference between the pixel in the centre of the applied window and the next pixel in scanline direction.

Table 5.1: Parameters' values (MSD approach).

Filter size	53 m
Height threshold	3 m
Slope threshold	30°

Regarding the NGPs approach, the first vertical threshold is based on the height difference between the ground points selected as a minimum from different scanlines. The second vertical threshold is based on the vertical altitude difference between the initial DTM and the input DSM. Higher values of this parameter lead to capturing higher regions positioned between the terrain and off-terrain regions. For the generation of the nDSM mask for both methods and sites, a threshold of two meters was applied (i.e. only objects with a height of more than two meters are highlighted).

Table 5.2: Parameter values (NGPs approach).

Filter size	53 m
Vertical threshold for accepting ground points	1.1 m
Vertical threshold for detecting ground mask	0.4 m

5.2.2 Qualitative evaluation

Area 1: Figure 5.4 displays DSM derived from ALS data (a) and a reference DTM (b), respectively.

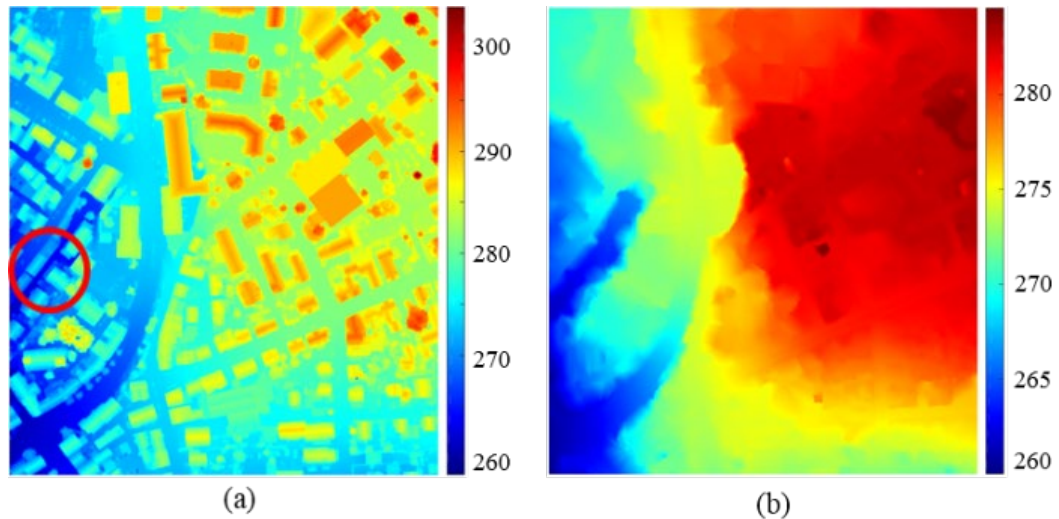


Figure 5.4: Area 1 DSM derived from ALS data (a) and reference DTM (b) (Mousa et al., 2017).

The DTMs created using the MSD approach and the proposed NGPs algorithm are shown in Figure 5.5. The first row shows the input DSM derived from image matching (a) with 14 cm GSD, the selected ground points (b), and the initial DTM (c). The second row shows the extracted ground mask (d), DTM (e), and nDSM mask (f) using the MSD algorithm while the extracted ground mask (g), DTM (h), and nDSM mask (i) using the NGPs algorithm are shown in the third row. The extracted ground masks are represented by the white areas. The extracted MSD ground masks (Figure 5.5(d)) shows that details are lost using the MSD method, as highlighted with yellow circles. In contrast, these larger regions belonging to the bare-ground are successfully identified by the NGPs approach (Figure 5.5 (g)). The NGPs approach is able to segment buildings, high vegetation, and even some cars, and then exclude them from the ground mask. In the resulting DTMs of both methods, there is one area which causes problems for the MSD as well as the novel NGPs method. This area is highlighted by the red circles in Figure 5.5(e) and (h). For this area, there is no actually error in the generated DTM, but instead it is actually due to an error in the input DSM. The error could be caused by occlusion or mismatch in the DSM generation which can

be seen in Figure 5.5(a). At the same time, in the LiDAR DSM in Figure 5.4, ground regions within the highlighted area are clearly visible.

In the extracted nDSM masks from both methods (MSD and the new NGPs method), the black regions in Figure 5.5(f) and (i) represent off-terrain pixels, i.e. buildings and high vegetation. The differences are clearly visible and are highlighted by the yellow circles, especially between Figure 5.5(d) and Figure 5.5(g). While many elevated bare-ground regions were lost by the MSD approach leading to a cluttered nDSM mask (as shown in Figure 5.5(f)), the nDSM mask created by NGPs is more realistic with respect to finding the building locations, as seen in Figure 5.5(i). This is mainly due to the successful identification of the network ground pixels as can be observed in Figure 5.5(b).

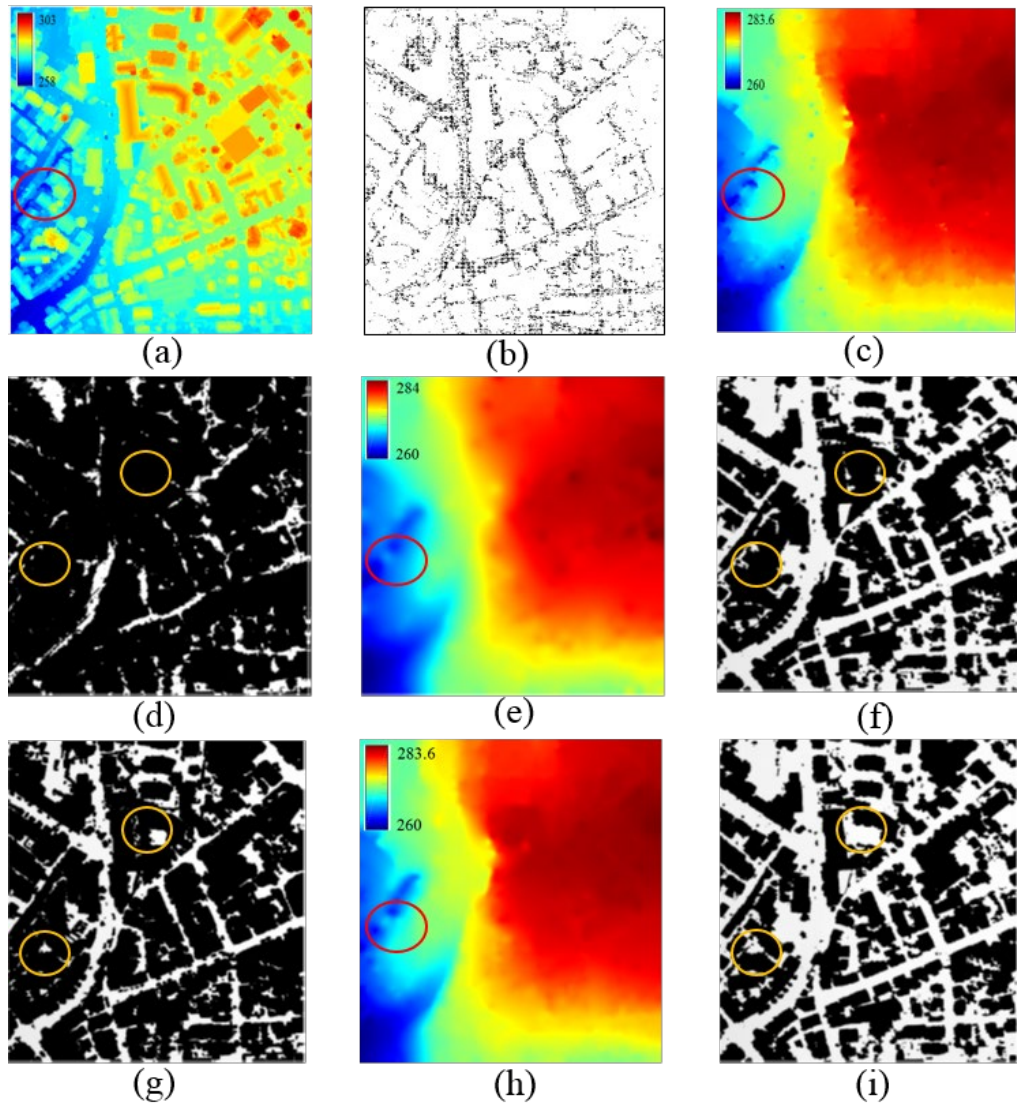


Figure 5.5: MSD versus NGPs DTM extraction for Area 1(Mousa et al., 2017). (Top) Common input:- DSM derived from image matching is shown in (a), with a 14 cm GSD, as well as the selected ground points (b), and the initial DTM (c). (Middle) MSD algorithm: - The extracted ground mask (d), DTM (e), and nDSM mask (f). (Bottom) NGP algorithm: - extracted ground mask (g), DTM (h), and nDSM mask (i).

Area 2: The DSM derived from LiDAR data and the reference DTM are shown in Figure 5.6(a) and (b), respectively. In Figure 5.7, the results of the MSD and the novel NGPs algorithms are given. The difference in the created ground masks of the MSD approach (Figure 5.7(d)) to the NGPs approach (Figure 5.7(g)) is clearly noticeable. For instance, wide ground regions have been lost from the ground mask created by the

MSD approach and non-ground points are clustered together. In contrast, the proposed algorithm NGPs successfully identifies those areas as can be seen in Figure 5.7(g).

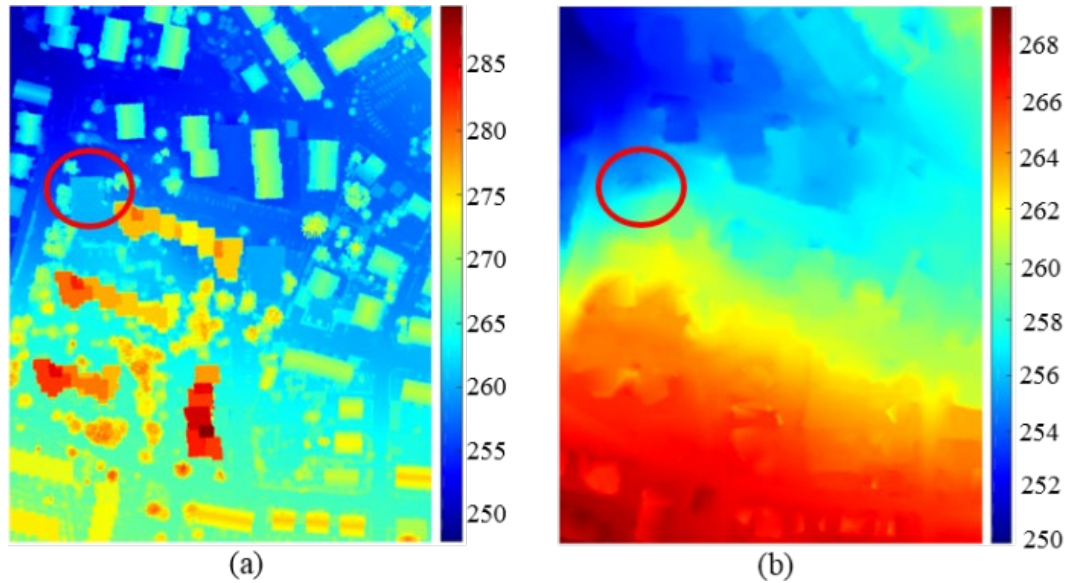


Figure 5.6: Area 2 DSM derived from LiDAR data (a) and the reference DTM (b) (Mousa et al., 2017).

The extracted DTMs created by the MSD and the NGPs are presented in Figure 5.7(e) and (h), respectively. Both DTMs seem to be similar except the areas highlighted by the red circles. It can be seen that the NGPs DTM values in this area are higher than the DTM created by MSD, and even higher than the LiDAR DTM, as highlighted in Figure 5.6(b). In fact, the true height values (the same values in the DSM because it is terrain area) for this ground area are higher than for all the created DTMs, as visible in the LiDAR DSM Figure 5.6(a). This means that the DTM created by NGP is the closest to the actual in this highlighted area.

In spite of the significant improvements in the quality of the ground masks created by the NGPs approach, the created DTM masks from both methods look similar (Figure 5.7(f) and (i)). This is because the topographic surface of Area 2 has an approximately constant slope. The one difference visible in the ground region is highlighted by the red circle. In this ground region, the height difference is up to 6 meters with a sudden change. Such a case is very difficult because large height changes are normally used to distinguish between terrain and off-terrain regions. However, a smaller part in this

area is incorrectly classified as non-ground region by NGP when compared to MSD (see Figure 5.6(i) and (f) respectively).

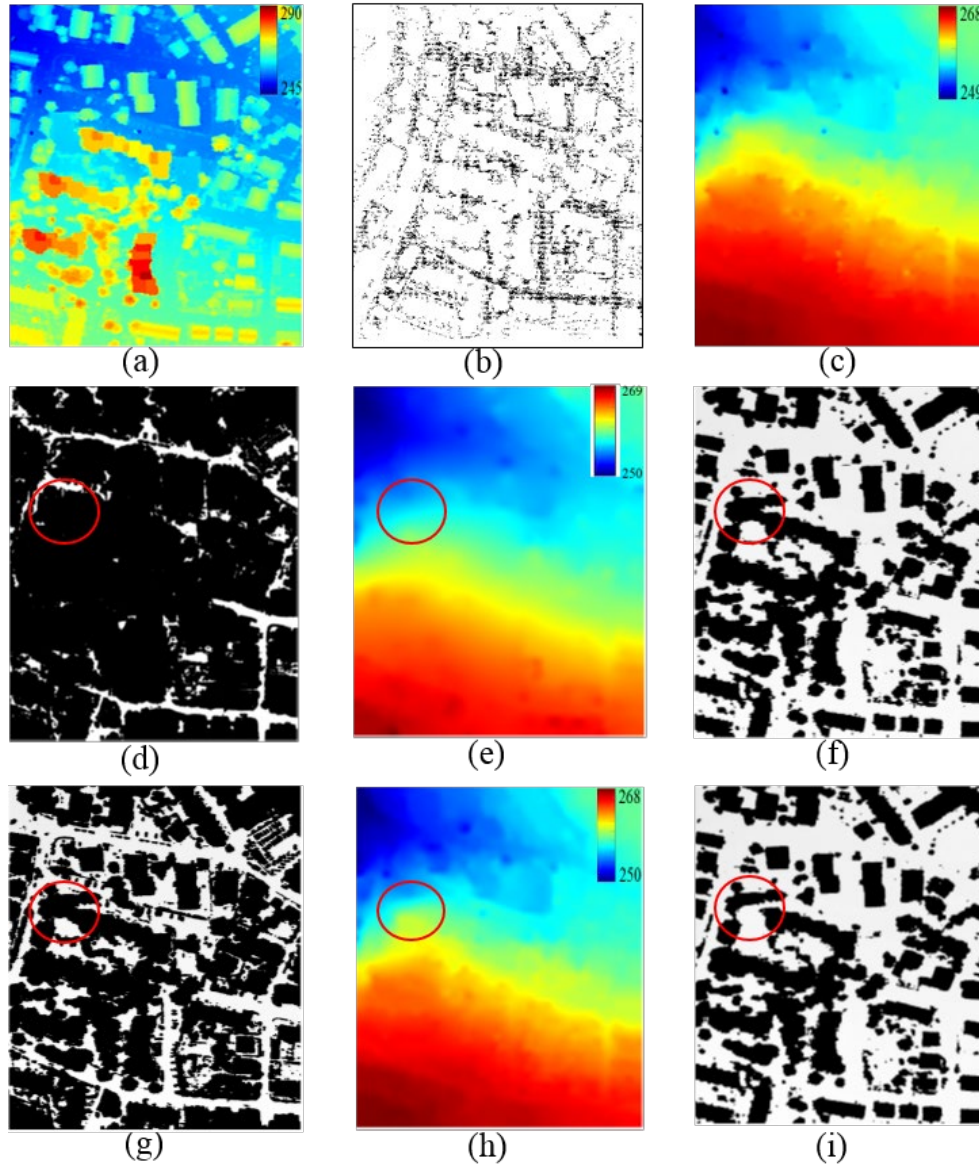


Figure 5.7: DTM extraction for Area1(Mousa et al., 2017). It shows the input DSM derived from image matching techniques (a) with 14 cm GSD, the selected ground points (b), and the initial DTM (c). The second row presents the extracted ground mask (d), DTM (e), and nDSM mask (f) using the MSD algorithm. The extracted ground mask (g), DTM (h), and nDSM mask (i) using the NGPs algorithm are shown in the third row.

This is possibly due to the procedure of comparison of the height difference between a point in the centre of a scanline and the point having the lowest elevation (located at

the end of the scanline because of its sloped area). This provides a high elevation difference while both the point in the centre and the lowest point belong to the terrain. Therefore, details were lost utilising the MSD method. This limitation was overcoming in the NGPs method through the creation of an initial DTM from the seeding of well-distributed ground points. The elevation of each pixel in the initial DTM was then compared to the elevation of the same pixel in the DSM to make a classification decision considering the height differences.

5.2.3 Quantitative evaluation

Area 1: The created reference DTM is subtracted from the DTMs created by the MSD and the NGP methods, and the resulting difference maps are shown in Figure 5.8(a) and (b) respectively. For both difference maps, positive differences mean that the created DTM is higher than the reference DTM. In some spots, the height difference reaches up to five meters (red areas), which is quite considerable. However, the error inside of the areas highlighted by the red circles is related to the error in the original DSM as discussed earlier. The second error, which is marked by red arrows, is related to the interpolation technique used in both methods. While inward interpolation has been applied in the MSD and NGP methods, LAsTools uses a standard linear interpolation on the DTM from LiDAR points.

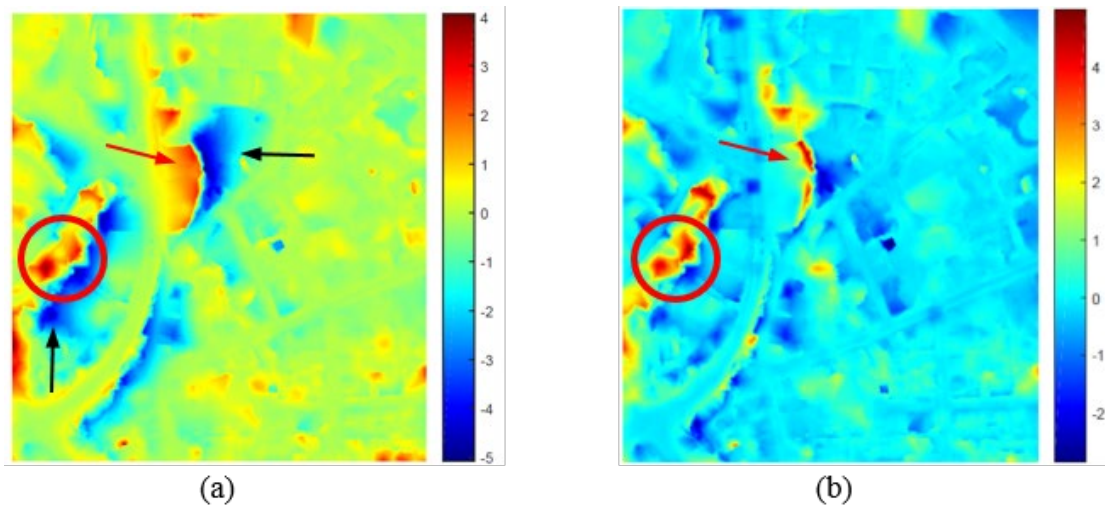


Figure 5.8: Area 1 elevation difference maps of the LiDAR DTM compared to the MSD (a) and the NGPs (b) (Mousa et al., 2017).

Huge negative height differences are highlighted by the black arrows and are presented in the DTM created by the MSD approach (in Figure 5.8(a)). Those differences are up to five meters and, compared with the two meters from the NGPs approach, have to be flagged as gross errors. The major reason for this is that the raised ground regions are lost from the ground mask due to the applied slope parameter. For instance, slight height differences between adjacent ground pixels within the 14 cm GSD lead to the wrong classification decision (see section 4.3.2), especially in raised terrain. Hence, those raised ground pixels are lost from the DTM created by the MSD approach and incorrectly classified as off-terrain pixels. Based on the difference of the DTMs and after excluding the gross errors, the mean-square error (MSE) and the standard division are computed and presented in Table 5.3.

Table 5.3: Area 1 statistics of Mean Square Error (MSE) and standard deviation (SD) of the height differences of MSD and NGPs compared with the LiDAR DTM as well as the time required to execute the algorithms.

Method	MSE (m)	SD (m)	Computation time
MSD	0.8560	0.3660	556.06 s
NGPs 4d	0.4574	0.2666	50.19 s
NGPs 8d	0.3814	0.2492	52.60 s

For further evaluation, the NGP algorithm was used with different numbers of directional scanlines applied: 4 (NGP 4d) and 8 (NGP 8d). While the NGP algorithm should be run with 8 directions, 4 were also used in this experiment in order to evaluate to what extent the NGP improves the results compared to the MSD approach by only using a different method of determining the ground initially. Hence, it can be analysed what the impact the method has on the success of detection ground points and their distribution. The mean squared error declines from the MSD to the NGP 4d, and then further to the NGPs 8d. Therefore, it can be concluded that the selection of the ground points improves the results. However, the introduction of additional scanlines seems to have a higher impact on the drop of the mean squared error. This conclusion is also verified when looking at the standard deviation. The computation time required to execute the NGP algorithm is significantly less than the MSD algorithm due to the reduced complexity as discussed previously. The used computer was a desktop machine (Processor: Intel(R) Core(TM) i7-4790 CPU @ 3.60 GHz and Installed Memory (RAM): 16.00 GB with 64-bit operation system).

As mentioned previously, ground truth for the DTM was not available, instead a reference classification map (layers including buildings, trees, grass, bare-ground, and cars) was provided. The extracted DTMs were compared with the reference classification map bare-ground layer. To do this, an nDSM is created by subtracting the final DTM from the input DSM, with threshold set at 0.4 metre. The extracted bare-ground mask is compared to the reference bare-ground layer. The grass layer contains regions belonging to bushes and short grass that can be consider as part of the terrain.

Thus, it is excluded from the extracted bare-ground mask. For this reference DTM the calculated completeness and correctness are 66% and 94% for the evaluation result presented in Figure 5.9(a). The yellow parts represent bare-ground areas that are correctly detected. The regions belonging to the roads are detected correctly most of the time as shown in Figure 5.9. The red parts indicate false positive errors due to those off-terrain regions being incorrectly classified as ground regions. The blue areas indicate false negative errors due to missing parts from terrain. Some parts actually belong to off-terrain regions, but they were incorrectly classified in the reference provided, as shown in Figure 5.9(d). The red plus signs in Figure 5.9(d) belong to the building layer as it has a height of 5.864 metres in the nDSM. Other missing parts are very difficult to detect due to a sudden change in elevation. Figure 5.9(b) presented an example, in which the DSM (red line in the profile) has a sudden change of nearly six metres height (from 274 metres to 285 metres) in a horizontal distance of less than 1 metre. Alongside, the DTM (blue line in the profile) was smoothed, thus parts from ground regions have been lost from the extracted ground regions. Even considering this, the NGP method showed a good performance in which the majority of cars were successfully eliminated from the terrain, as can be seen in Figure 5.9(c).

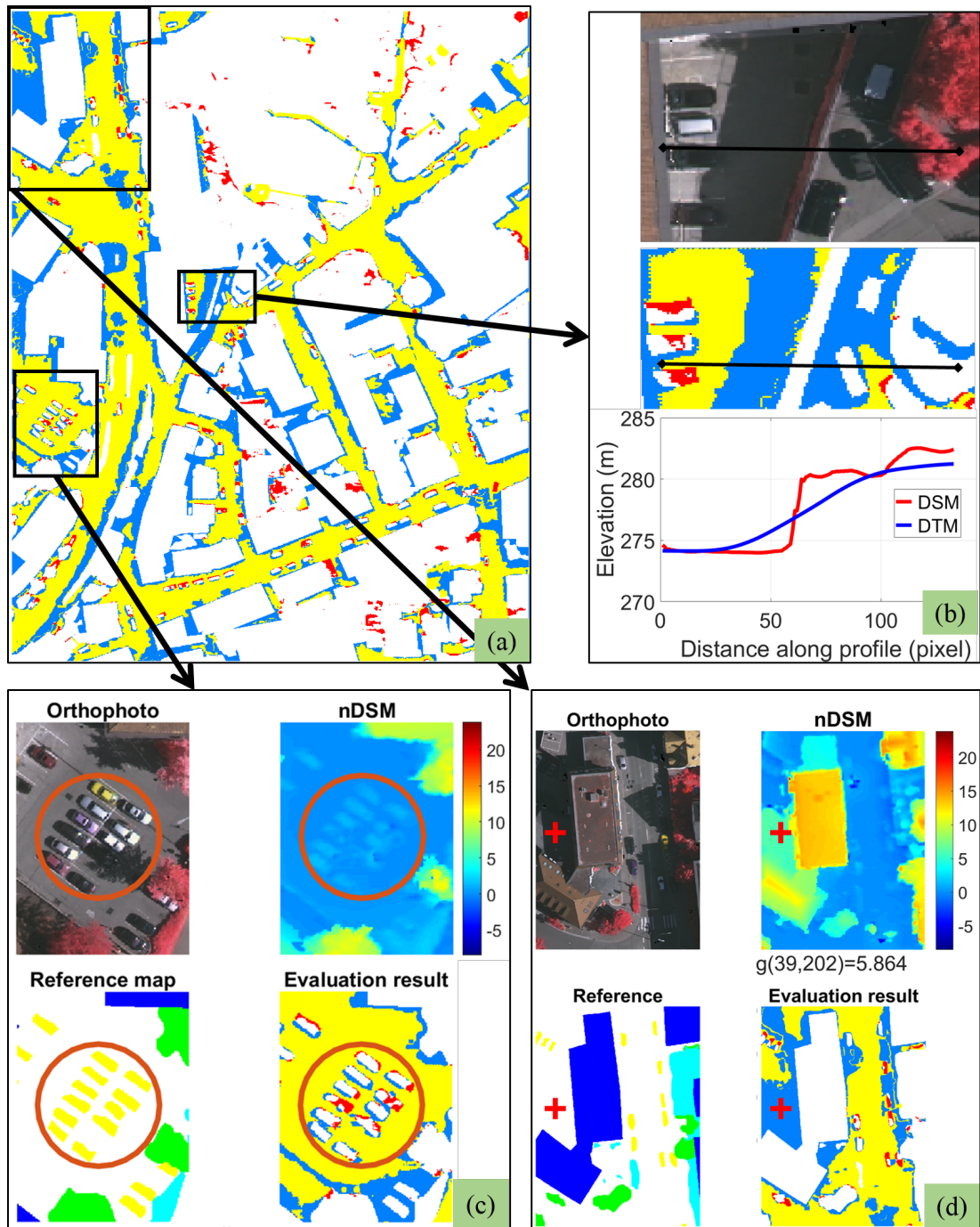


Figure 5.9: The DTM evaluation result and analysis for NGPs method. The evaluation result is depicted in (a), where the true positive, false negative, and false positive are showed in yellow, blue, and red respectively. Analysis of some misclassification are presented in (b) and (d), while (c) shows removed cars from terrain. In the reference map, buildings, trees, grass, bare-ground, and cars, are depicted in dark blue, green, cyan, white, and yellow, respectively.

Area 2: The elevation difference maps of the MSD method (Figure 5.10(a)) and the NGP method (Figure 5.10(b)) reference DTM compared to the are both significantly better than the DTMs created of Area 1. This is mainly because the topographic surface is less undulating and because there are no noticeable errors in the input DSM. The regions highlighted with the red circle in Figure 5.10(a) indicates large negative errors up to two metres in the MSD extracted DTM. The MSD approach is still facing the same challenge as highlighted and discussed previously based on the outcomes. In contrast, the maximum negative error (Area 2) in the DTM created by the NGP method is smaller by approximately 0.5 metre. While there are no significant lower sections in the NGP, there is one higher area by nearly 2.5 m, as surrounded by red circle Figure 5.10(b). This area is also highlighted previously in Figure 5.6(a). In fact, the correct value for this area is higher than what the NGP approach determines. Consequently, NGPs shows significantly better performance and therefore the DTM values are the closest to the true values.

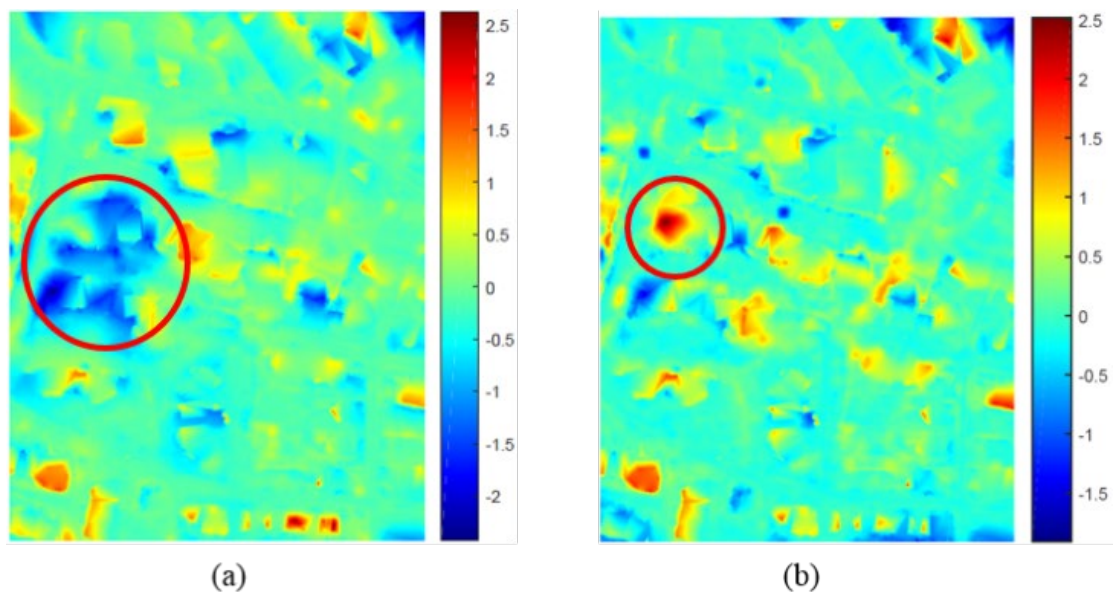


Figure 5.10: Area 2 height differences maps of the created DTMs with LiDAR DTM. MSD (a) and NGPs (b).

The calculated Mean Squared Error (MSE) and standard deviation (SD) of the height differences are presented in Table 5.4. The MSE and SD equal to 0.1942 metres and 0.1079 metres for the DTM created by MSD, 0.1488 metres and 0.0873 metres for (NGPs 4d), and 0.1348 metres and 0.0879 metres for (NGPs 8d) respectively.

Accordingly, the quality of the created DTM by the NGP method is slightly improved in Area 2 compared to Area 1 due to the fact that Area 2 is rather flat. However, the better quality achieved indicates that the NGP algorithm successfully overcomes the negative impact of implementation of the slope parameter in a high resolution DSM. Similar to Area 1, NGPs algorithm requires significantly less time as concluded previously.

Table 5.4: Area 2 statistics of Mean Square Error (MSE) and standard deviation (SD) of height differences between MSD and NGPs compared with the LiDAR DTM and time required to execute the algorithms.

Method	MSE (m)	SD (m)	Computation time (s)
MSD	0.1942	0.1079	5480.5
NGP 4d	0.1488	0.0873	319.47
NGP 8d	0.1348	0.0879	327.16

Similar to Area 1, in Area 2 the nDSM also uses a threshold at 0.4 metres to extract the ground regions. The calculated completeness and correctness are 87% and 77%, wherein the evaluation results are presented in Figure 5.11(a). The false positive error (red parts) is noticeably more than in Area 1. Many of them are related to tree borders where the nDSM provides a height value less than the applied threshold (0.4 metres). For instance, the black circle in Figure 5.11(b) shows the nDSM is about negative 0.003 metres, therefore is classified as being a ground region. Thus, it flagged as a misclassification. In Figure 5.11(c), false negative errors occurred caused by a sudden elevation change similar to Area 1 (Figure 5.9(b)).

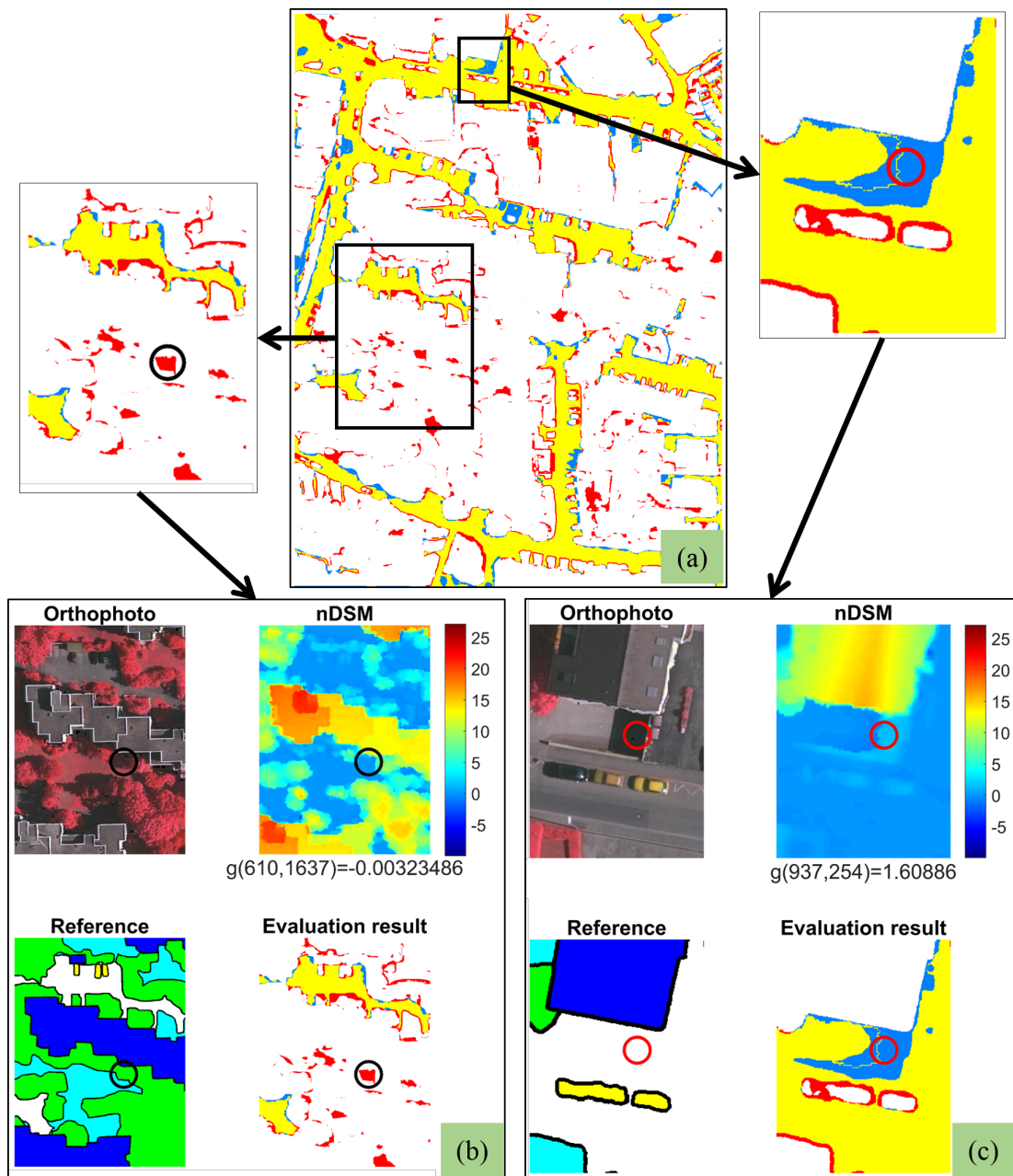


Figure 5.11: The DTM evaluation result and analysis for NGPs method. The evaluation result is depicted in (a), where the true positive, false negative, and false positive are showed in yellow, blue, and red respectively. Examples of some false positive and false negative are presented in (b) and (c). In the reference, buildings, trees, grass, bare-ground, and cars, are depicted in dark blue, green, cyan, white, and yellow, respectively.

5.3 Building segments extraction and evaluation

Building segments were extracted based on the proposed fusion method in section 4.4. Based on the literature, various evaluation methods have been carried out to assess classification results. Pixel or area-based methods, as well as object-based methods (Rottensteiner et al., 2014; Rutzinger, Rottensteiner, & Pfeifer, 2009) are used to calculate completeness, correctness, and quality. Object-based evaluation was performed two times; one test considering that the smallest existing building of interest has an area of at least 10 m^2 , while the second test assumes the smallest building to be 50 m^2 .

The proposed data fusing method (section 4.4) is evaluated firstly using the benchmarked Areas 1, 2, and 3 in Vaihingen. The proposed method is also evaluated using the Melville dataset utilising 126 buildings which were manually digitised as ground truth data.

Vaihingen dataset: The calculated completeness (C_m), correctness (C_r), and quality (Q_l) are shown in Table 5.5. The performance of this approach is depicted in Figure 5.12.

Table 5.5: Area-based and object-based evaluations of building outline extraction output. C_m = completeness, C_r = correctness, Q_l = quality.

Area	Area/ pixel-based (%)			Object-based					
				$>10 \text{ m}^2$			$>50 \text{ m}^2$		
	C_m	C_r	Q_l	C_m	C_r	Q_l	C_m	C_r	Q_l
1	92.70	96.50	89.70	93.80	100.00	93.80	100	100	100
2	95.20	94.80	90.50	91.70	91.70	90.00	100	100	100
3	87.10	97.40	85.10	87.50	100.00	87.50	97.4	100	97.4
Avg.	91.67	96.23	88.43	91.00	97.23	90.43	99.13	100	99.13

Concerning the area-based evaluation, the proposed method shows satisfactory results with a high average correctness of 96.23%, and an average quality of 88.43% (Table 5.5). Considering the minimum area of building objects which exist in the scene is larger than 50 m^2 , the average quality rate is 99.13%. The performance of the proposed

fusion method for the three Vaihingen sites is visualised in Figure 5.12. The red circles in Figure 5.12(c2) indicates a building with inadequate laser point data, while the gridded DSM presented better 2.5D information. This specific building was missed by several methods based on point clouds segmentation (Rottensteiner et al., 2014) due to insufficient information. Working on a gridded DSM can overcome this problem more proficiently, as shown by this building having been successfully identified, as highlighted by the black dashed circle in Figure 5.12(c).

Generally, there are two types of errors: false positives (red) and false negatives (blue). The false positive errors represent those objects identified as buildings, but are actually not buildings in the ground truth (e.g. trees or vehicles). It can be seen that the false positive errors are insignificant and mainly caused by misalignment errors. The false negative errors are buildings which were not detected. Only two low buildings were missing (i.e. in Area 3 Figure 5.12(c)). The misalignment of the fused data could also yield false negative alarms. Thus, it can be concluded that the proposed data fusion (see equation (4.4) has been successfully applied and, therefore, buildings are well-distinguished from trees, even those located in shadow regions. Thus, the robustness of the approach - that nDSM, NDVI, and planarity are sufficient to remove trees – is proven because few changes happen by modifying these thresholds by around 10%.

In addition, there are some challenging cases in which the proposed procedure struggles in its performance, which result in false negative errors. For example, those missing buildings in Figure 5.12(a1), Figure 5.12(b), Figure 5.12(c1), and Figure 5.12(c3). The problem is due to the relative elevation between those buildings and the DTM being too small. There are two reasons for this, either buildings are too low (i.e. building in Figure 5.12(b)) or the DTM (blue dashed line) spuriously too high (i.e. building in Figure 5.12(c3)). Profiles to explain this issue are depicted in Figure 5.12(b2), Figure 5.12(c1), and Figure 5.12(c4).

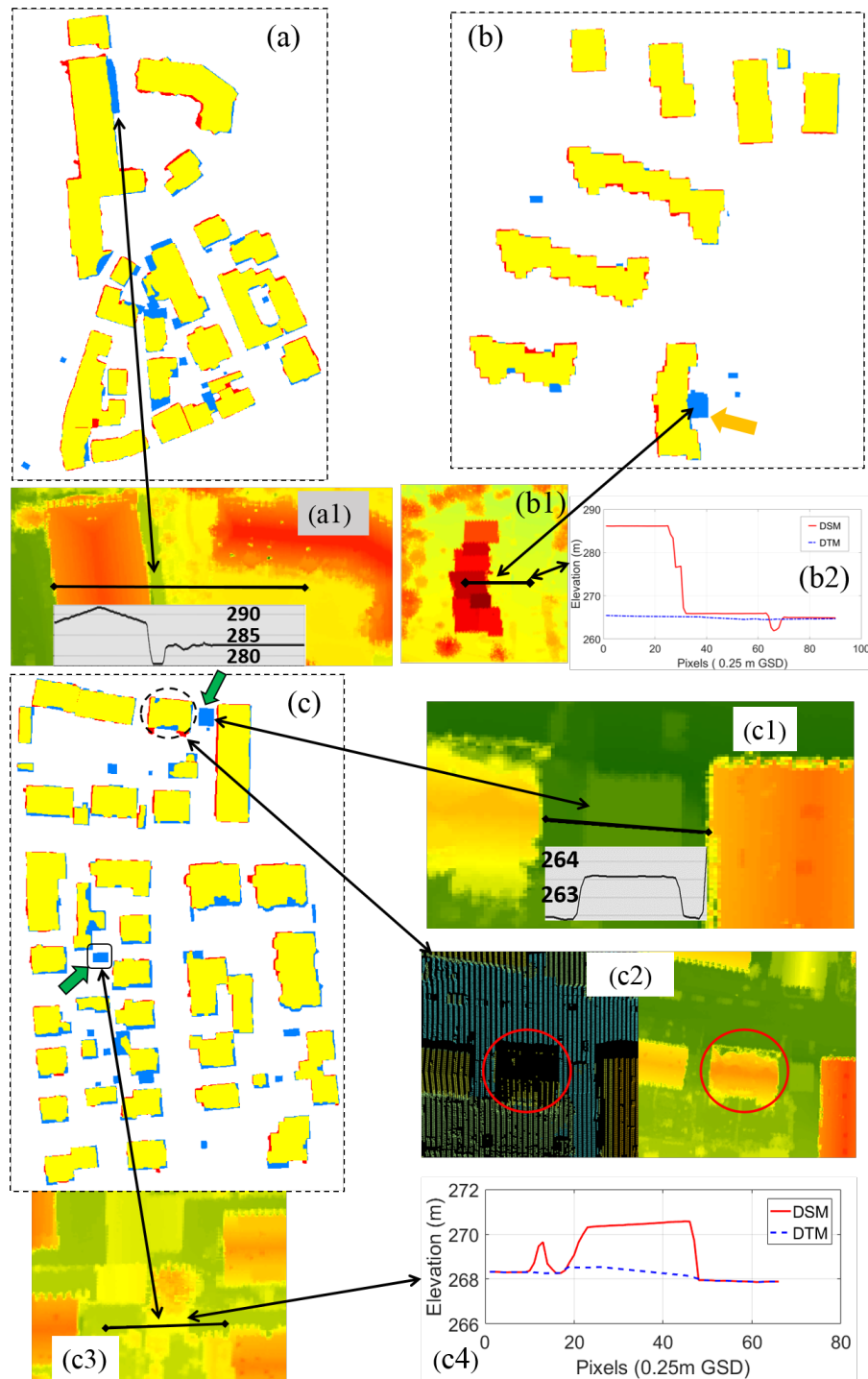


Figure 5.12: Building detection results in Vaihingen (Mousa et al., 2019): (a) Area 1, (b) Area 2, and (c) Area 3. True positives, false negatives, and false positives are depicted in yellow, blue, and red respectively.

In comparison with the results presented in Rottensteiner et al. (2014), the proposed method shows a quite good performance for the area-based evaluation with an average

quality of 88.43%. It is also positioned within the highest level in terms of correctness, with an average of 96.23%. Compared with the methods based on morphological filtering by (i) (Mongus et al., 2014) and (ii) (Zongze Zhao et al., 2016), the proposed method shows a higher average quality by 2.6% and 3.4%, respectively, over the three sites, but a lower completeness in Area 3 by 1.8% and 0.6%, respectively. In addition, the proposed method shows a significantly higher area-based quality rate for Area 1 by 4.8% and 6.2% over sources (i) and (ii), respectively. This is an indicator that morphological filters show a worse performance in sloped terrain. In comparison with the lidar data-fusion method (Du et al., 2017), the proposed method shows a lower average area-based quality by 1%, but a higher average correctness by 1.36% in the three areas.

Compared to point-cloud-based approaches such as Awrangjeb et al. (2014), the proposed method performed significantly better in area-based correctness and quality (96.23% versus 91%, and 88.43% versus 80.7%, respectively). Only Area 3 showed a lower completeness rate by approximately 0.6% compared to Awrangjeb et al. (2014). However, while that work assumed the DEM was available, this paper has presented a fully automatic workflow. In terms of geometrical analysis, the proposed method yielded a higher accuracy by 0.17m (0.75m versus 0.87 m).

Melville dataset: A ground truth for Melville dataset including 125 buildings was manually created by digitising outlines based on the ALS data and the orthophoto. Those 125 buildings are considered as reference and are used for the evaluation of the building detection and regularisation results. The calculated completeness, correctness, and quality for those 125 buildings are 94%, 95%, and 90%, respectively. The results are visualised in Figure 5.13. The findings highlighted in Vaihingen dataset are also demonstrated here. For example, false negative errors occurred in the building shown in Figure 5.13(d). This is mainly due to the low height difference problem, as shown in the nDSM in Figure 5.13(d) in which the elevation difference is nearly 1.7 metres. Also, the misalignment errors can also be seen in this data. The misalignment error is the main limitation of procedure based on fusion data from multiple sources and it is highly depending on how the orthophoto image was created.

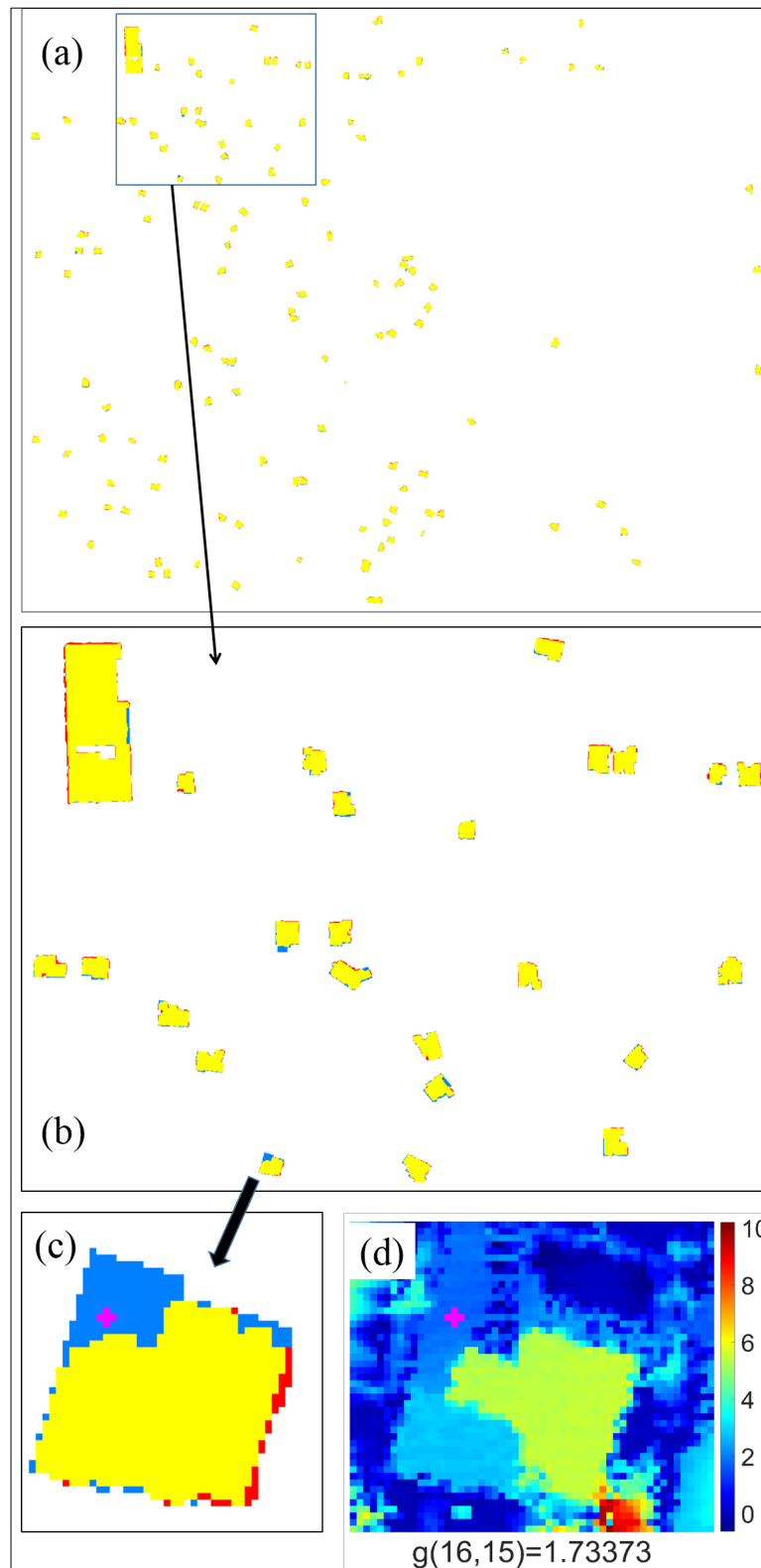


Figure 5.13: Building detection in Melville. (a) represents the result for 126 buildings. (b) enlarged section, while (c) and (d) shows one segment building and its nDSM.

It is important to mention that a shapefile with the building outlines for Melville from 2012 was available however the existing dataset was captured in 2016. Therefore, it is reasonable to investigate whether buildings were still existing, demolished or had been extended since 2012. To do so, the shapefile involving the building outlines from 2012 was rasterised first as shown in Figure 5.14(b), and denoted as an “*old map*” (i.e. out-of-date map). Then, the extracted building map from the existing data (2016) (see Figure 5.14 (c)) was overlaid with the *old map* generating an additional raster map (Figure 5.14(d) which indicates changes). Buildings in Figure 5.14(d) are categorised into three groups: unchanged (brown), new building segments (yellow), and demolished buildings (magenta). The unchanged buildings are those existing in both the old and new map without significant alterations. The new buildings are those existing in the new building map, but not in the old map. In contrast, the demolished buildings exist in the *old map*, but no longer existing in the new map. Further, for the new and demolished buildings, it also important to investigate whether those buildings have been changed in part or in whole. For instance, new buildings segments should be separated further into two groups: wholly new and partially new. Similarly, demolished buildings should be also separated into wholly and partially demolished.

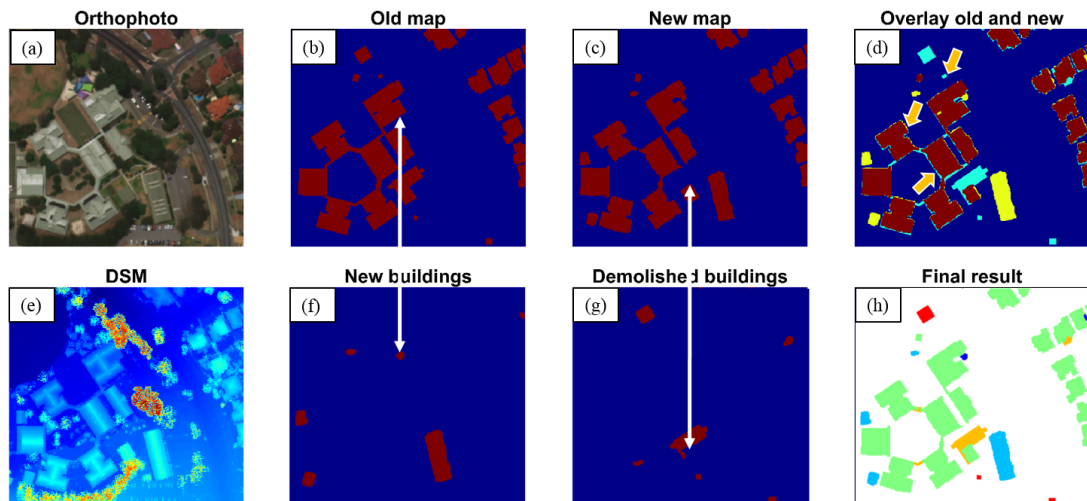


Figure 5.14: Procedure for updating buildings map. Orthophoto, old, and new buildings are shown in (a), (b) and (c), while the overlaid old and new map is presented in (d). The DSM is shown in (e). The new and demolished building segments are shown in (f) and (g). The final updated map is shown in (h).

Smaller segments having an area of less than 10 m² were removed. This is because buildings smaller than this are rare and those segments are most likely due to being noise. Moreover, misalignment errors between the two datasets can yield a false indication. Additionally, the width and length of a building should be at least three metres (Vosselman, Kessels, & Gorte, 2005). Thus, small segments not achieving these requirements are eliminated by analysing the connected components, the area, as well as the major and minor axis of each single segment. Figure 5.14(d) shows some examples of those small and misaligned segment to be removed as indicated by the orange arrows. The new and demolished building segments are presented in Figure 5.14(f) and (g) respectively. In order to differentiate between completely and partially changed buildings, a simple and efficient procedure is proposed. First, each individual new and demolished building segment is dilated by one pixel. Then, a new building segment is classified as building-part if an existing overlap between its border (the dilated pixels) with a building in the *old map* Figure 5.14(b), otherwise it is classified as a completely new as indicated by white arrows as in Figure 5.14. While, a demolished building segment is classified as demolished-part if its border overlaps with a building in the new map as in Figure 5.14(c), otherwise it is classified as completely demolished as can be seen in Figure 5.14(f).

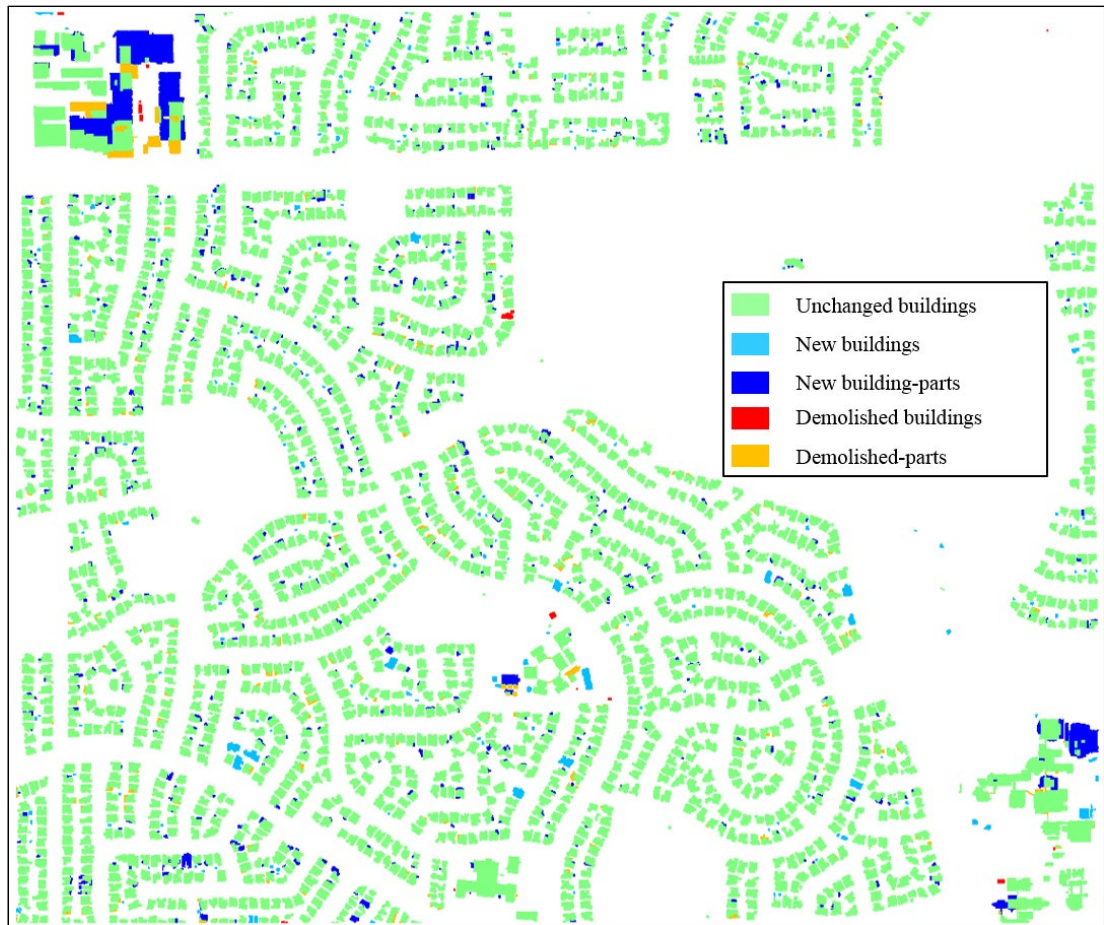


Figure 5.15: Final updated building map for the Melville study area. Unchanged, new, new part, demolished, and demolished-parts buildings are presented in brown, dark blue, light blue, orange, and light green respectively.

The final updated building map for the Melville dataset is presented in Figure 5.15. It can be seen in this figure that the updated map has several changes, i.e. many new and extended buildings, completely demolished ones, etc. By visual inspection of many of these changes, it can be seen that the majority of the proposed changes are correct. However, in some cases, there is missing data from the input DSM which leads to false errors of demolished buildings and building-parts. Moreover, wholly new buildings are sometimes classified incorrectly as new building-parts. This is because the new building is connected to its neighbours making them appeared to be connected as shown Figure 5.16 (new map). The buildings coloured in magenta in the final resulting map (Figure 5.16) demonstrate such incorrect classification. This classification error could be analysed further by considering a rule-based procedure. For example, by

analysis of the intersecting region, which is very small compared to the actual building outlines, can be used to identify joined building outlines which should be separated. However, this is not the focus of this thesis and therefore, is not pursued here.

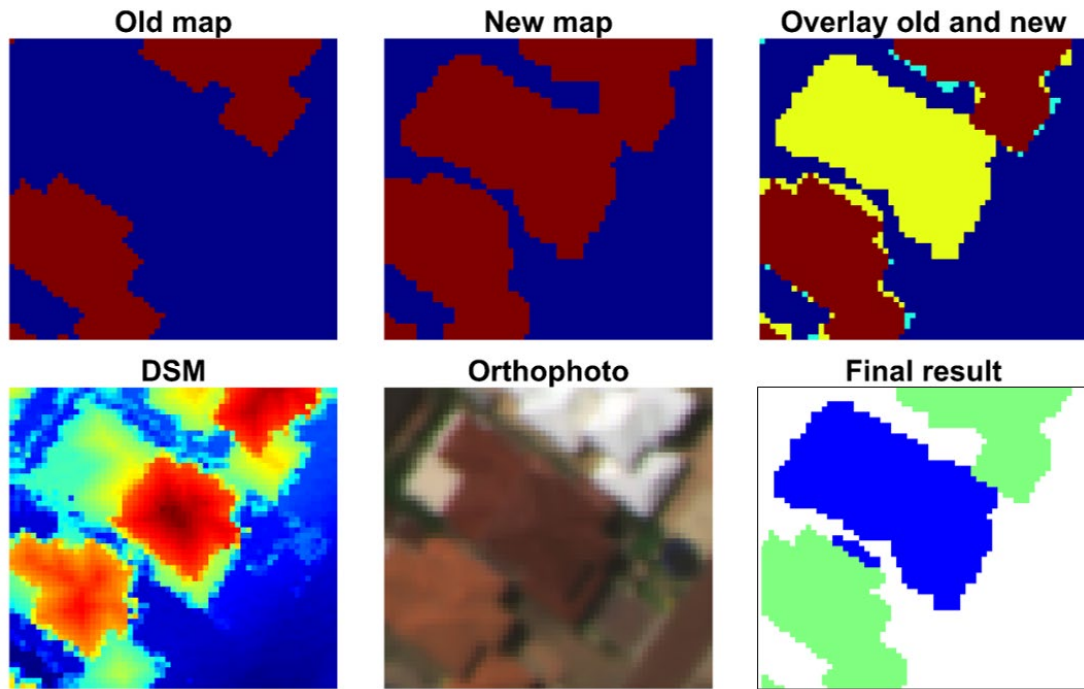


Figure 5.16: Example of a complete new building classified incorrectly as a new building-part. The old map, new map, and their overlapping values are shown in the top row. The DSM, Orthophoto, and the result for the updated map from left to right are presented in the bottom row.

5.4 Evaluation of the proposed data-driven approach for building polygon generalisation

This section aims to evaluate the performance of the proposed simplification algorithm for the generation of approximated building polygons (see section 4.5.2). To do so, a comparison with the Douglas-Peucker (DP) method is conducted. Sensitivity analysis on algorithm parameters is also presented.

5.4.1 Comparison to the *Douglas-Peucker* (DP) algorithm

The DP algorithm was chosen as a comparison because it is the most common simplification algorithm in the literature as discussed previously. In order to provide a fair comparison, the parameters of both algorithms were adjusted, so that the same number of vertices and line segments were acquired. As the number of line segments is identical, the standard deviation (SD) of the orthogonal distances between the input boundary points and their corresponding line segment is a reasonable measure to compare the success of both methods during the evaluation process. An algorithm that achieves a lower number of vertices associated with a smaller SD or RMSE value indicates a better performance. This is because simplifying a shape with less vertices or edges, alongside with best fitting those edges with the input boundary points, will consequentially achieve a smaller RMSE.

First, it is highlighted that the DP algorithm is very sensitive to the position of the starting point. Figure 5.17 illustrates the results of the building generalisation depending on the start point location (black triangles indicated by the orange arrows) for both methods. In this figure, boundary points are shown by coloured dots assigned to their corresponding edges. The top row presents the DP algorithm results and the bottom row presents the results of the proposed method. While the DP algorithm produces significantly different type of outputs regarding the shape and number of obtained vertices (despite of a fixed tolerance threshold value (T) with value of 10 pixels, with a pixel = 0.25 m), the proposed method shows a more stable performance. The same number of vertices is obtained with a fixed value of the RMSE threshold (RT) equal to 7 pixels. Hence, where possible, it is desirable to fix the number of vertices. However, this is not possible using the DP algorithm. The main reason is that the tolerance threshold (T) corresponds to the farthest boundary point to its assigned edges. This farthest point is not always located at a building corner, which is the only way to detect vertices in the processing of the DP. In contrast, corner preservation has been implemented in the new simplification approach through the proposed likelihood equation (see section 4.5.2). The proposed method utilises the RMSE value as a threshold, which is more robust against outliers than the DP method because the outliers can be hidden in the calculated RMSE value.

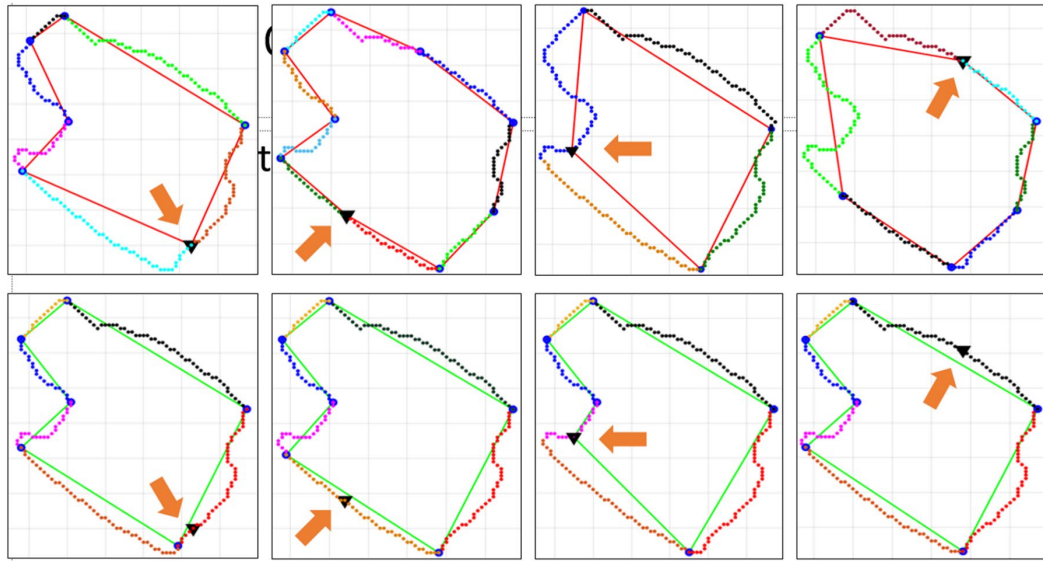


Figure 5.17: The sensitivity to a change of the starting point for the DP algorithm (top row) and the proposed method (bottom row) (Mousa et al., 2019).

In the DP-based algorithm, the final number of vertices changes from 6 to 9 to 4 and finally 6 (from left to right) even with one fixed tolerance threshold (10 pixels). The proposed algorithm seems more stable as the same number of vertices (6) has been obtained using one RMSE threshold value equal to 7 pixels.

Figure 5.18 presents the results for comparison utilising different building shapes processed by the proposed algorithm and the DP-based method, applying different parameters for both algorithms.

The parameters are:

- for DP, the tolerance threshold (T) (defining the distance from the furthest boundary point to its related line segment); and
- for proposed algorithm, the RMSE threshold (RT) (defining the RMSE of all distances between boundary points belonging to their corresponding line segment).
- the standard deviation values for both algorithms.

Examples 1 and 2 shows the same input building boundary points that should be represented by 6 vertices and 6 edges to achieve the best fit. However, the best possible DP simplification output is 7 vertices with a SD value of 2.2 pixels. Reducing the

vertices number to 6 by changing the value for T (from 16 to 19 pixels), it diverges the shape away from the optimal solution and produces a SD value of 3.8 pixels. In contrast, the proposed algorithm achieves 6 vertices and a SD value of 0.89 pixels. Other performance issues are related to the ad hoc tuning parameters of both algorithms when dealing with buildings of different complexity. For instance, simple building structures like the third example in Figure 5.18 requires a minimum T value of 8 pixels to produce 4 vertices while more complex shapes (example 6 in the figure) requires a $T \geq 11$ pixels.

(example 6 in in Figure 5.18) requires a $T \geq 11$. In addition, for other types of buildings (e.g. example 7 and 8) it is a long iterative process to determine the T value yielding the best outcome. In contrast, the proposed algorithm requires significantly less empirical tuning for the RT value; a RT value of 4 pixels is suitable for all types of buildings presented in Figure 5.18. Overcoming such issues is essential for automation of building regularisation, especially if buildings with different complexity levels exist in a scene.

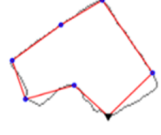
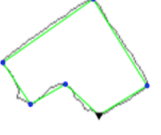
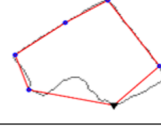
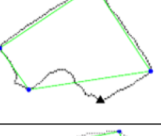
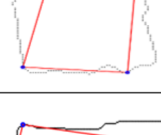
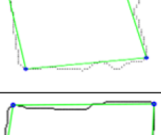
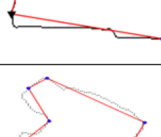
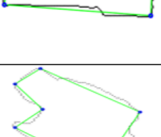
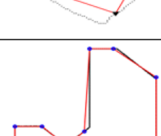
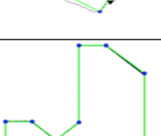
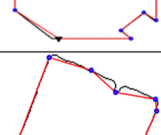
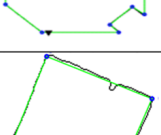
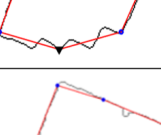
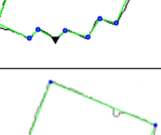
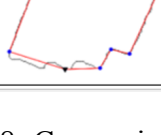

Example	DP	T (threshold)	V	SD	Proposed	RMSE (threshold)	V	SD
1		9-15	7	2.20		4-9	6	0.96
2		16-19	6	3.86		10-15	5	3.63
3		8-30	4	3.20		4-18	4	1.17
4		21-37	4	5.71		4.4-27	4	1.26
5		10	6	2.60		4.2-20	6	1.35
6		11-23	13	2.56		3-27	13	0.67
7		6	12	1.45		4	10	1.03
8		7	9	1.76		5-6	6	1.44

Figure 5.18: Comparison of simplification results generated by the DP algorithm and the proposed algorithm (Mousa et al., 2019). Triangles denote the starting boundary point. T is the tolerance threshold for DP. V is the number of vertices (\bullet), SD is the calculated standard deviation. Note the RMSE threshold is used in the proposed algorithm.

5.4.2 Parameters setting and analysis for building generalisation

The goal of this section is to find reasonable ranges for the following parameters: nDSM threshold (NT), Median filter (MF), RMSE threshold (RT), Angle detector (AD), and Similarity Ratio (SR). For the sensitivity analysis, five different values for each of the applied parameters were used, i.e.

- nDSM threshold (NT): 1.5, 2, 2.5, 3, and 3.5 metres.
- Median filter (MF): 0, 0.5, 1, 1.5, and 2 metres.
- RMSE threshold (RT): 0.4, 0.8, 1.2, 1.6, and 2 metres.
- Angle detector (AD): 1, 1.5, 2, 2.5, and 3 metres.
- Similarity Ratio (SR): 100, 90, 80, 70, and 60%.

While the nDSM threshold NT affects the results of the building detection, it has no significant impact on the regularisation algorithm results. Typically, the optimum selection of the nDSM threshold varies in the range of two to three metres depending on the minimum height of man-made buildings in the scene. Figure 5.19 shows that increasing the NT threshold leads to an increase in the correctness measure and a reduction of the completeness measure. This is expected because small buildings and building parts are remaining undetected.

In the regularisation step, a sensitivity analysis was performed for the proposed approach. The RMSE of the extracted vertices is calculated to estimate the planimetric accuracy of the regularisation results. In Figure 5.19(b), the numbers 1–5 along the x -axis indicate the set values for each of the five parameters. The y -axis shows the calculated RMSE of the extracted vertices. When one of the parameters was changed, the other parameters were held at their standard values as follows: $MF = 1$ m, $RT = 1$ m, $AD = 2$ m, and $SR = 90\%$.

Overall, the RT parameter was found to be the most sensitive. Choosing a suitable value is highly dependent on the complexity of buildings in the scene and on the desired degree of simplification. For example, Area 2 (Figure 5.20) has complex buildings with many small edges (sometimes less than two metres in length) which required a strict RT value of 0.75 m (Table 5.6) to preserve these details. The utilised

RT was set higher (1.1 m Table 5.6) for Area 2 compare to Area 3. Therefore, some small details have been lost (Figure 5.20(c)). Such small details can be retained with a tighter RT threshold (similar to Area 2), but the regularisation goal is to obtain the highest reduction in number of boundary points, while at the same time the original building shape should be maintained as much as possible.

The MF and AD parameters show less sensitivity in the behaviour and both can be fixed to default values of 1m and 2 m respectively (see Table 5.6). Similarly, the SR shows a more robust behaviour and only at a value of 60% did the calculated RMSE increased from 0.75 to 0.83 metres. At 60% the chance for a building to be oversimplified and modelled as a rectangular is too high.

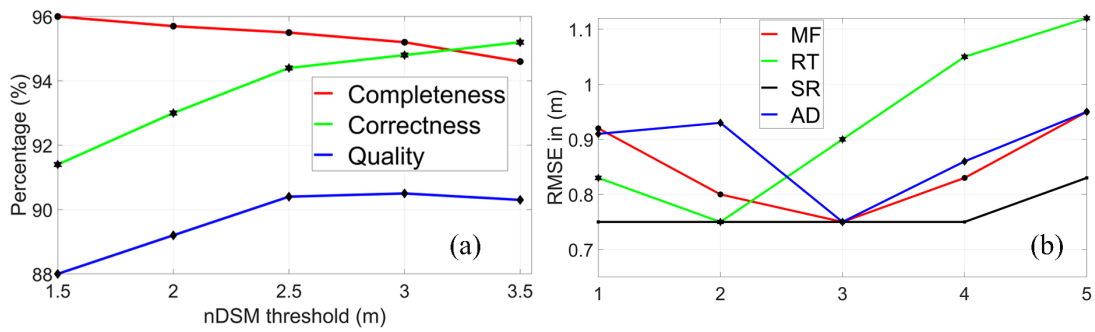


Figure 5.19: Sensitivity analysis of the parameters (Mousa et al., 2019). (a) sensitivity analysis of NT threshold. (b) sensitivity analysis of median filter (MF), RMSE threshold (RT), angle detector (AD), and similarity ratio (SR).

The optimal selection of applied parameters depends on the input data resolution, minimum height of the buildings, the desired level of details and the degree of the simplification. Table 5.6 summarised the parameters used for all study areas. Some parameters are invariant between data sets (i.e. the angle detection parameter AD), while some are not applied (i.e. the median filter in the Melville dataset because of its low resolution).

The Melville dataset was found to contain buildings placed quite close to each other, and applying the median filter was found to merge adjacent buildings to a single complex shaped building, causing the regularisation result to become less reliable

(Mousa et al., 2019). Instead, a post-processing step involving morphological filtering (*erosion* followed by *dilation*) with a kernel of three pixels was used on the extracted building segments to enhance the outlines. After the generalising of the approximate building polygons according to these parameters, the next step regularised or enhanced the approximated polygons through applying the GMM and GHM adjustment.

Table 5.6: Parameters setup for building regularisation.

	<i>MF (m)</i>	<i>NT (m)</i>	<i>RT (m)</i>	<i>AD (m)</i>	<i>SR %</i>
Area 1	1	2.3	1.4	1.75	75
Area 2	1	2.7	0.75	2	~
Area 3	0.75	2	1.1	2	68
Melville	~	2	0.9	2	85

5.5 Buildings Regularisation Results and evaluation

Those polygons are either processed by GMM or GHM. Rectilinear building polygons are processed by GHM while non-rectilinear ones are processed by GMM (see sections 4.6.2 and 4.6.3). It is important to mention that the GHM processes only polygons having an even number of vertices. Thus, if a rectilinear building introduced to the adjustment with an odd number of vertices, the GHM cannot be applied, and therefore, GMM is used.

In this section, the regularised building outlines after adjustment using GMM or GHM (as applicable) are presented and assessed. The modelling assumptions applied to rectilinear and non-rectilinear buildings are also evaluated. Finally, an investigation of how the GMM and GHM adjustment improves the geometric accuracy of building outlines is performed.

5.5.1 Quantitative evaluation of buildings regularisation

Building regularisation results for the Vaihingen (Areas 1, 2, and 3) and the Melville datasets are presented in Figure 5.20 and Figure 5.21 respectively. The calculated RMSE values for the evaluation (see section 4.7) requires two corresponding

polygons. These two polygons are the extracted and regularised building outline, and its reference. Hence, if two adjacent buildings are extracted as a single (merged) outlines, the calculated RMSE becomes meaningless and does not reflect the absolute similarity between the extracted polygons and its reference polygons. This issue is demonstrated in the Vaihingen dataset where many adjacent buildings are merged together into a single segment. Hence, the quality of the building outlines in Vaihingen are assessed by a tool provided by ISPRS used in Rottensteiner et al. (2014). In this tool, the perpendicular distances between each vertex of an extracted building polygon and the nearest boundary points from the reference is calculated. Distances greater than a pre-defined threshold of three metres are excluded (Rottensteiner et al., 2014). Accordingly, the RMSE of the extracted vertices and the RMSE of building object centroids as well as the processing time, are presented in Table 5.7. Additionally, building polygons are rasterised, and then the completeness (C_m), correctness (C_r), and quality (Q_l) are also calculated. The rasterisation process may introduce errors, but it provides a robust outcome (Potůčková & Hofman, 2016)

Table 5.7: Evaluation results of building reconstruction: area-based completeness (C_m), correctness (C_r), and quality (Q_l), RMSE of the extracted vertices and RMSE of centroids of building objects as well as the processing time in seconds.

Area	<i>Area or pixel based (%)</i>			<i>RMSE of</i>	<i>RMSE of</i>	<i>processing</i> <i>time in sec</i>
	C_m	C_r	Q_l	<i>vertices</i> (m)	<i>centroids</i> (m)	
1	88.7	95.6	85.3	0.93	0.80	97.69
2	93.4	95.3	89.3	0.75	0.57	67.71
3	84.1	96.1	81.4	1.04	0.70	94.05
Avg.	88.73	95.67	85.33	0.91	0.69	

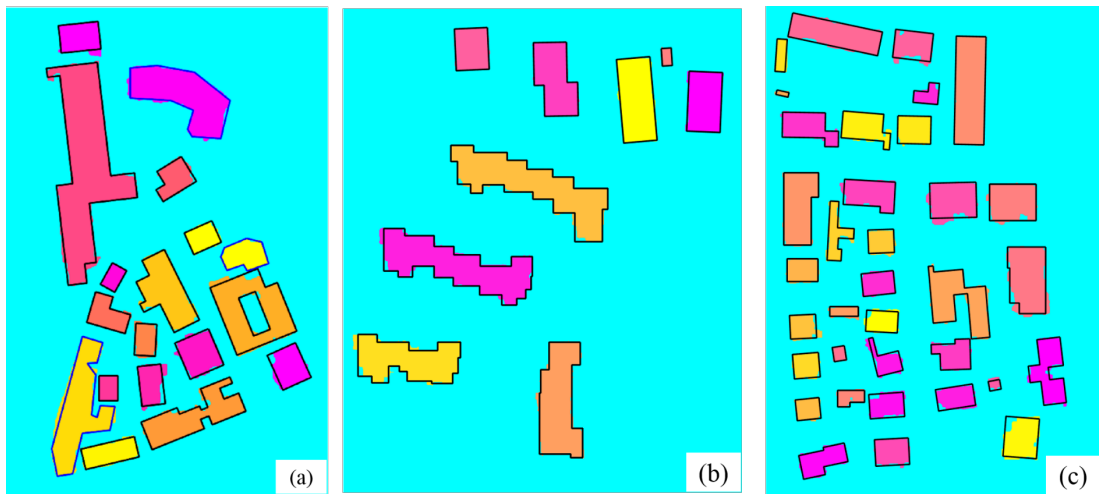


Figure 5.20: Building regularisation result in Vaihingen dataset (Mousa et al., 2019). Areas 1, 2, and 3 from left to right. Building colours are randomly assigned for better visualisation.

In Figure 5.20, it can be seen that the majority of the complex buildings have been modelled accurately, especially in Area 2 (Figure 5.20(b)). In Area 2, the achieved RMSE values are 0.75 metre for the extracted vertices and 0.57 metre for centroids of building objects. A high rate of correctness and a high rate of completeness have been achieved for all three sites with an average of 95.67% and 88.73%, respectively. The average RMSE of the geometric accuracy is 0.91 metre and the RMSE of the building centroids is 0.69 metre in all areas. The RMSE seems to be slightly higher in Areas 1 and 3. However, it is anticipated that the RMSE increases with a higher degree of generalisation and vice versa. For instance, adding more polygon vertices or edges usually leads to a lower RMSE. However, this is not the goal of the simplification process.

In regard to the Melville dataset, the proposed methodology was applied to the whole study area, including approximately 2200 buildings as presented in Figure 5.21. However, the 125 buildings manually digitised and used for the purpose of building segments evaluation are utilised only for the regularisation assessment. One of these 125 buildings has an inner polygon and thus, gives 126 polygons in total to be evaluated.

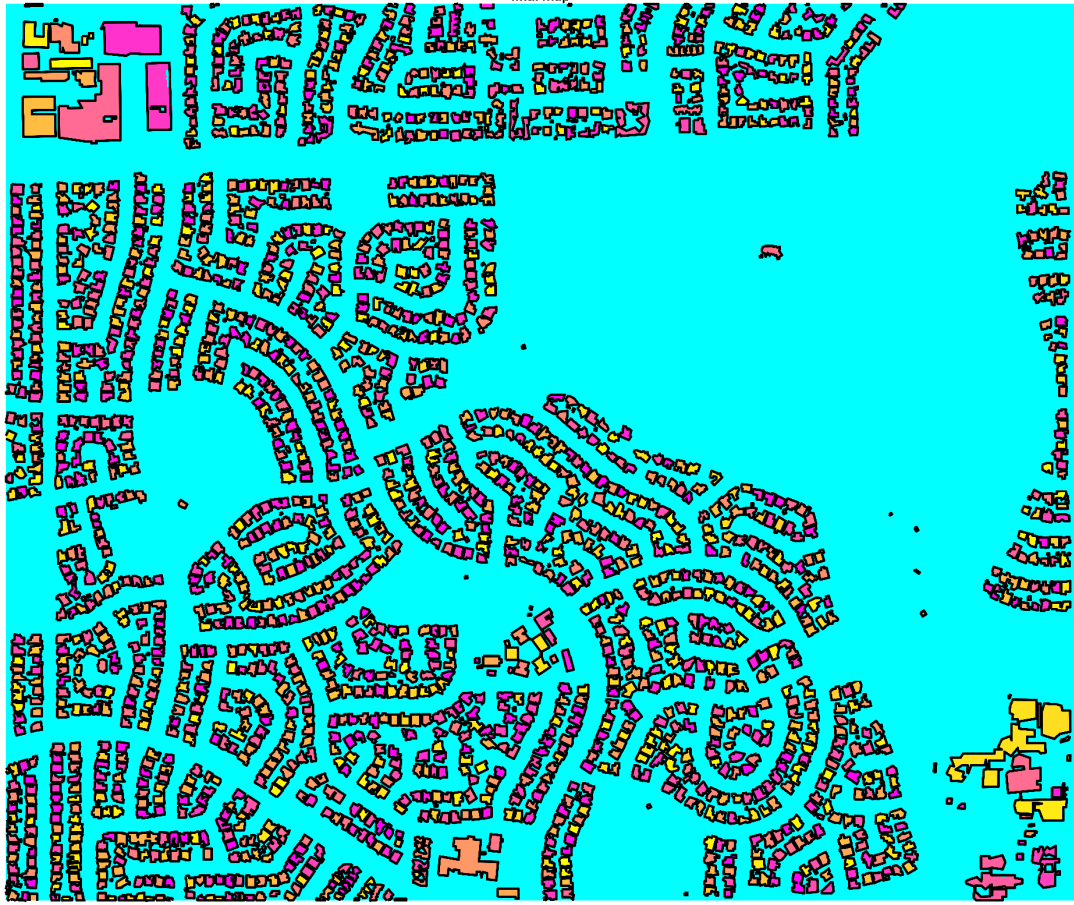


Figure 5.21: Building regularisation result in the Melville dataset. Building colours are randomly assigned for better visualisation.

The absolute RMSE of each extracted polygon and its reference is calculated in pixels (see equation (4.25)). Additionally, the centroids shift between all the extracted polygons and their references are calculated. The average RMSE of the extracted polygons before and after the adjustment are 2.03 pixels and 1.83 pixels respectively. The comparison of the RMSE for each building before and after the adjustment is presented in Figure 5.22. This comparison is represented as percentage improvement in Figure 5.23. Accordingly, the adjustment has achieved approximately 9.7% improvement in the calculated RMSE. It can be seen in Figure 5.23 that the accuracy of majority of those building polygons has been significantly improved. There are some polygons where decreased accuracy was presented caused by introducing inaccurate approximate polygons to the adjustment. Analysis and details will be

provided in the next sections including the model assumption evaluation and the assessment of the applied GMM and GHM.

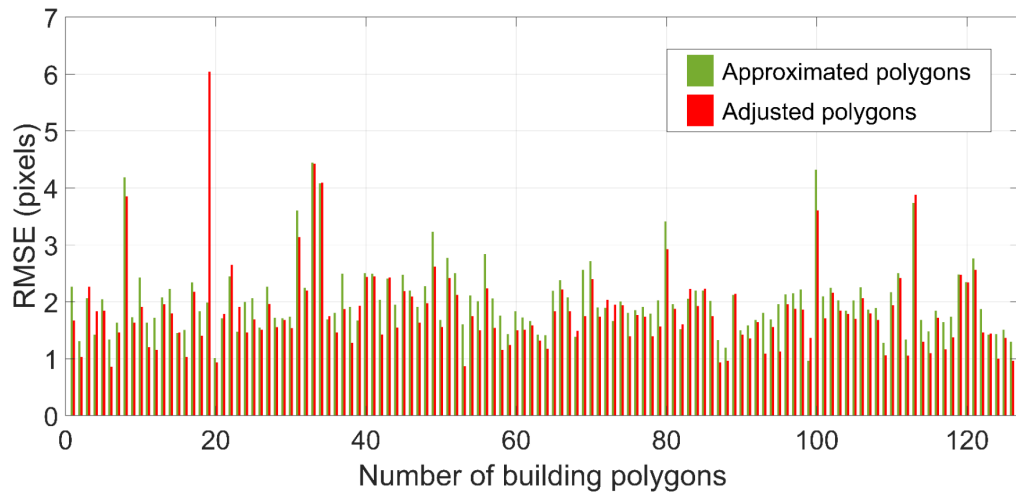


Figure 5.22: The calculated RMSE in pixels of the extracted polygons before and after the adjustment – Melville dataset.

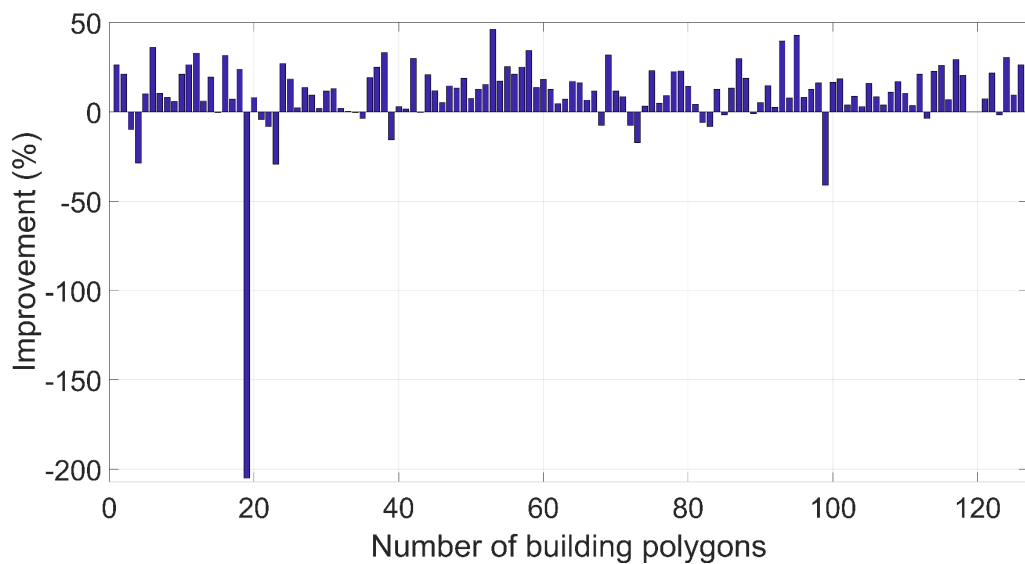


Figure 5.23: Improvement of the RMSE after the adjustment – Melville dataset.

5.5.2 Model assumption evaluation

For the evaluation, the Melville dataset was chosen for two reasons. Firstly, the low resolution DSM and the orthophoto both had a 0.5 metre ground sample distance (GSD). Secondly, the dataset includes rectilinear and non-rectilinear buildings with very complex outlines. Both of these factors presented the proposed methodology with a number of challenges. The ground truth involved 87 rectilinear buildings and 39 non-rectilinear buildings, giving 126 buildings in total.

Rectilinear and non-rectilinear buildings are assumed to be modelled using the GHM and GMM adjustments, respectively. The accuracy of this assumption is evaluated in Table 5.8. This evaluation shows that only 29 out of 39 non-rectilinear buildings have been processed by the GMM adjustment. In addition, less than half of the rectilinear buildings were processed by the GHM adjustment. In the next sections, the reasons some buildings were processed inaccurately are discussed.

Table 5.8: Model assumption evaluation – Melville dataset

	No. of buildings in the reference	No. of building modelled by GMM	No. of building modelled by GHM	Comp. %	Corr. %
Non-rectilinear buildings	39	29	10	74	39
Rectilinear buildings	87	45	42	48	81

5.5.3 Assessment of the GMM adjustment

As mentioned earlier, the GMM is designed to process non-rectilinear building polygons. It is important to mention that, for rectilinear buildings, if the approximated polygon is introduced to the adjustment with odd number of vertices, GHM cannot be applied. Therefore, such rectilinear buildings are modelled using GMM. This is because the GMM is free of constraints, which make it flexible for processing any

building shape (i.e. rectilinear or non-rectilinear). This section is divided into two subsections includes rectilinear and non-rectilinear buildings processed by the GMM.

5.5.3.1 Rectilinear buildings processed by GMM adjustment

As described in Table 5.8, 45 rectilinear buildings were processed by GMM. The majority of those building polygons have been significantly improved as shown in Figure 5.24(a) and (b). The GMM method does struggle for modelling some complex building polygons which were inaccurately approximated by the proposed data-driven method (section 4.5.2). For instance, if the positions of the estimated vertices were not close enough to the actual locations or missing vertex\vertices (especially building with too small edges i.e. less than one metre), an improvement might not be achieved in the adjustment.

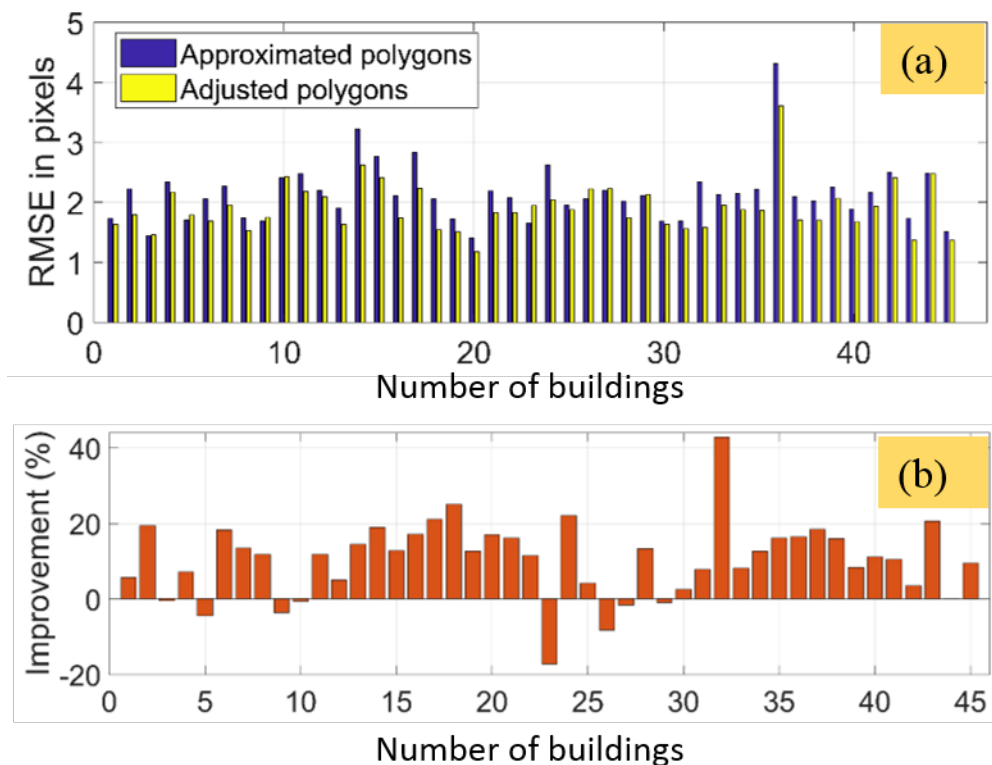


Figure 5.24: Assessment of rectilinear building polygons processed by GMM – Melville dataset (a) shows the RMSE of the approximated and adjusted polygons. (b) presents the percentage improvement of the RMSE planimetric accuracy.

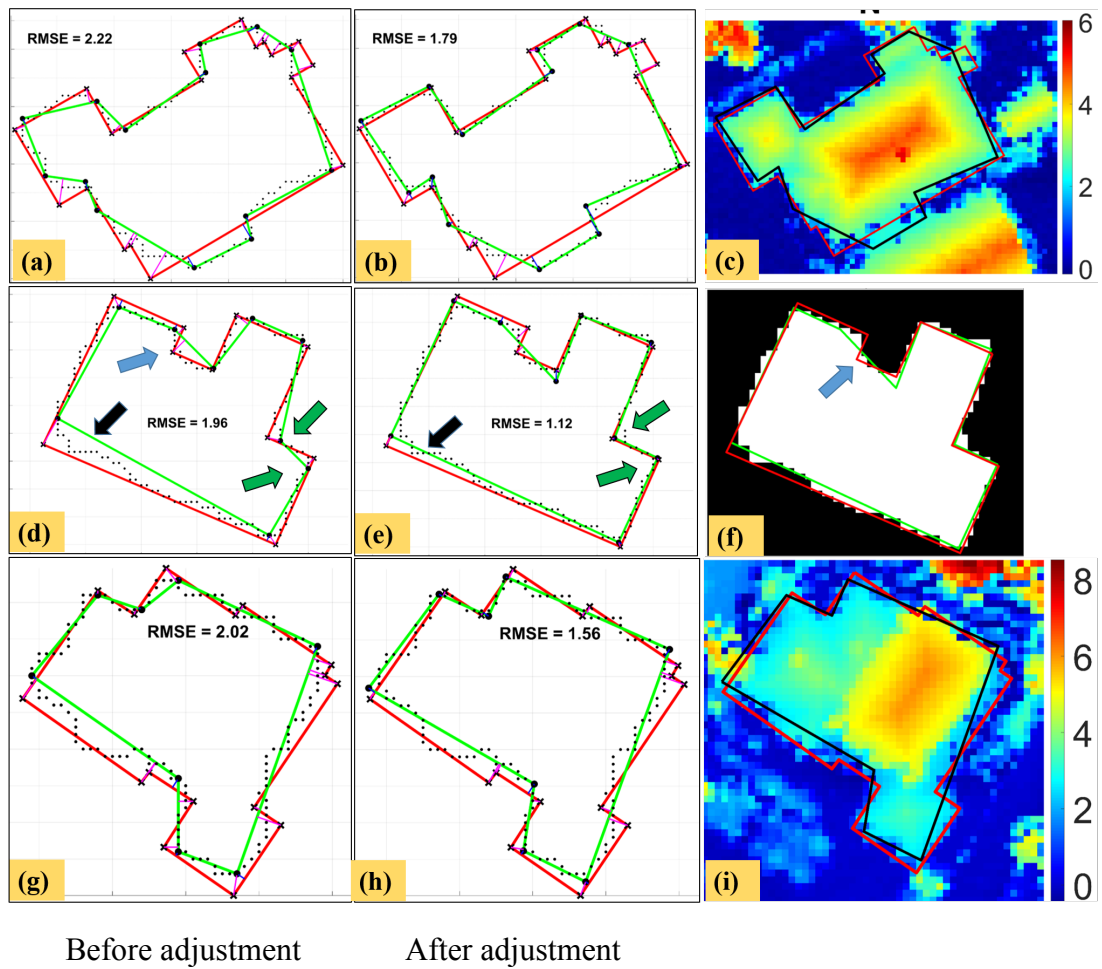


Figure 5.25: Examples show improving planimetric accuracy of rectilinear buildings processed by GMM. First column (a), (d) and (g) depict the approximated polygons (green) and the reference (red polygons) before adjustment. Second column (b), (e) and (h) represent the final extracted polygons after adjustment (green) along with references. The third column present nDSM and binary image.

Figure 5.25 illustrates three different examples for rectilinear buildings processed by the GMM adjustment taken from the Melville dataset. The first example (top row) presents a complex building with some short edges less than one metre. Such small details are very hard to extract with a low-resolution DSM (0.5 metres). Moreover, nDSM shows that this building is connected to its neighbour, resulting in a noisy border as shown in Figure 5.25(c). The middle row show a less complicated building, but there was an undetected vertex (as indicated by the blue arrow Figure 5.25(d)) due to it being located at nearly a straight line (not corner), as shown in the input building segment (Figure 5.25 (f)). It has been highlighted that any missing vertices in the

generalisation process can lead to a decreased building area in the final adjustment result.

Similarly, the third example (third row) presents a building with very small edges (less than a metre). While for buildings with odd number of vertices, GHM adjustment cannot be applied, the applied GMM adjustment performs very well in terms of the best fitting of polygon edges with the input boundary points as indicated by the black arrows on Figure 5.25(d) and (e). In terms of creating right angles, the GMM adjustment also performs very well, as indicated by the green arrows in the same two figures. The planimetric accuracy improved by nearly 42% after the adjustment. Similar to the first example, the third example (third row) presents a building with very small edges (less than a metre). Also in this case, the GMM adjustment has achieved a significant improvement.

In Figure 5.26, two examples of rectilinear buildings processed by GMM are presented in which the RMSE has enlarged after adjustment. In the top row, the nDSM shows a different shape compared to the reference due to an installed shade cloth next to the building. The dark green shade cloths is clearly visible in the image taken from Google Maps (Figure 5.26(a1)). Analysing the angles, GMM is able to create approximately right angles, especially when the input boundary points are approximately right angles. For instance, as indicated by the blue arrow in Figure 5.26(b), this right angle was correctly modelled even though the initial approximated angle in Figure 5.26(a) was not a right angle. This led to this vertex moving further away from the reference, which resulted in a larger RMSE of the adjusted polygon compared to the approximated polygon. Overall, the main reason for the larger RMSE after adjustment is due to a non-building feature which is not in the reference, but which has been identified by the proposed method successfully.

The second example in the second row of Figure 5.26 ((c) and (d)) presents another complex building outline with very short edges. Many of those short edges have been lost due to the necessary threshold decisions. The extracted building segment (Figure 5.26(c1)) shows that these short edges are part of the initial solution, but have been

merged in the approximated polygon. While the approximated polygon could achieve a RMSE value equal to 1.66 pixel, the adjusted polygon's RMSE value increased by nearly 17%.

Next to eliminated short edges, another contributing factor to an increase in RMSE after adjustment is that one vertex moved away from its reference (indicated by the blue arrow in Figure 5.26(d)), while its original approximated location was closer to the ground-truth. This is because the GMM adjustment tends to improve the best fit of edges rather than keeping the adjusted vertices close to the initial solution. Compared to the edges, the weight of the vertex locations (observation equation of vertices) is the identity matrix. Additionally, it can be seen in Figure 5.26(d) that one vertex is missing just to the right of the blue arrow. The adjustment is unable to recover the missing/inaccurate input data.

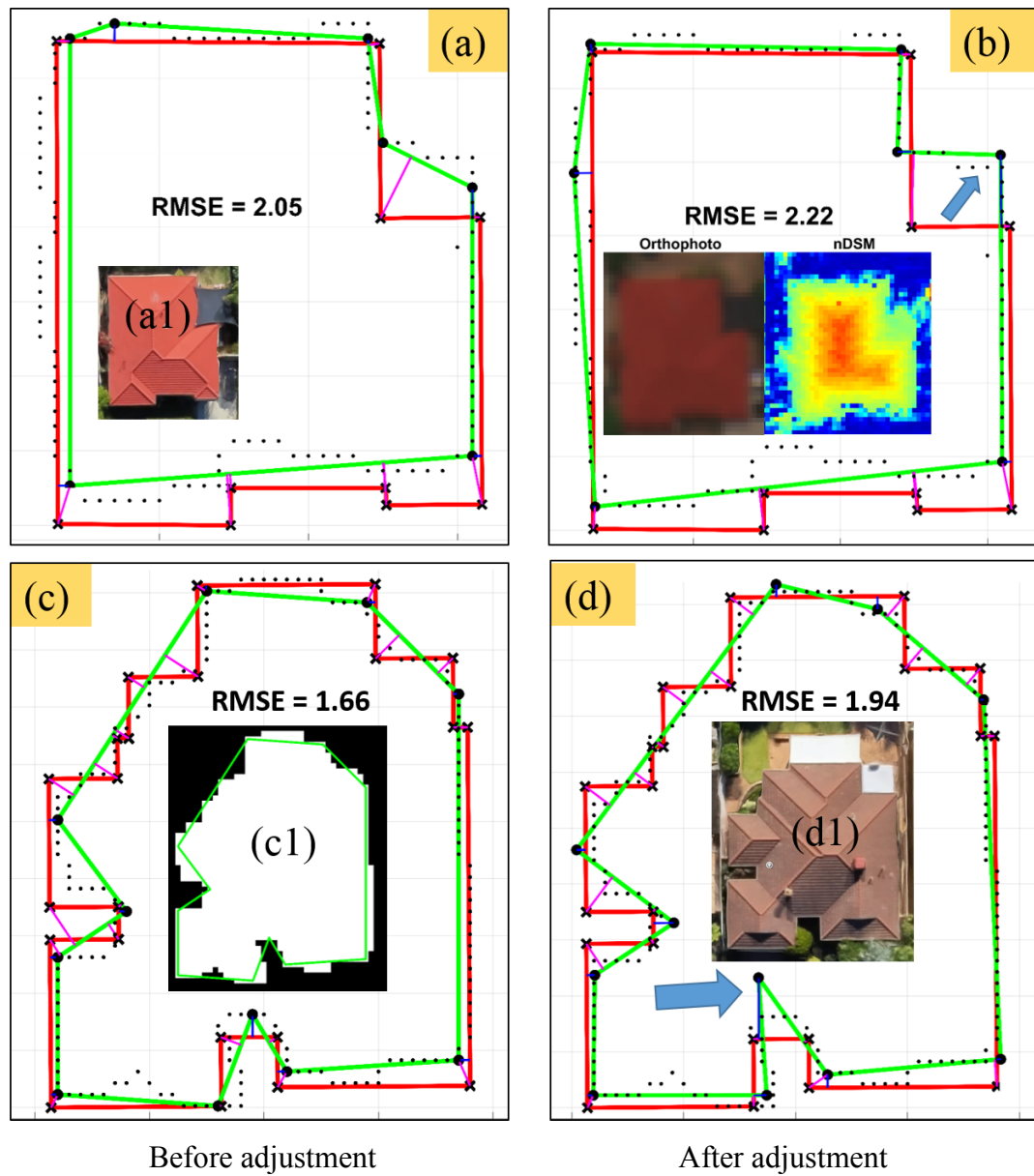


Figure 5.26: Examples show the declining planimetric accuracy of rectilinear buildings processed by GMM. First column (a) and (c) depict the approximated polygons (green) and the reference (red) polygons before adjustment. Second column (b) and (d) represent the final extracted polygons after adjustment (green) along with references (red). (a1) and (d1) are images taken from Google Maps while (c1) is the input building segment as extracted from the nDSM.

5.5.3.2 Non-rectilinear buildings processed by GMM adjustment

The aforementioned evaluation concerned on rectilinear buildings processed by the GMM adjustment. In this section, the performance of the GMM will be assessed as applied to non-rectilinear buildings, where it performs best. Table 5.8 shows that 74% of non-rectilinear buildings have been processed correctly by the GMM adjustment. The planimetric accuracy of the majority of those buildings improved after adjustment as can be seen in Figure 5.27. The only cases where deterioration was observed related to inaccurate input data (i.e. DSM or inaccurately approximated polygon).

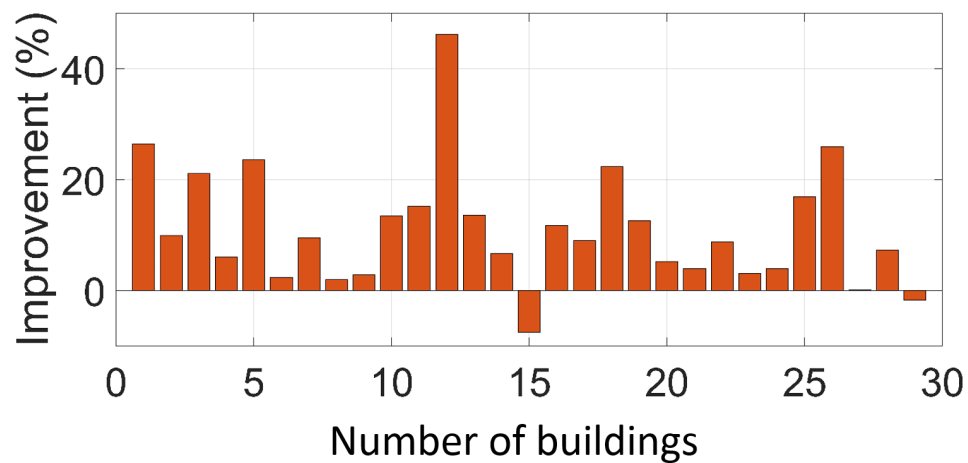


Figure 5.27: Assessment of non-rectilinear building polygons processed by GMM presenting the percentage of the improvement of the RMSE planimetric accuracy.

Figure 5.28 illustrates three different non-rectilinear buildings where the RMSE has significantly decreased which means the geometric accuracy is improved. For instance, Figure 5.28 (a) shows that the initial polygon (green) has a higher RMSE value compared to the adjusted result in Figure 5.28 (b) (also green). This is mainly due to the significant movement of the vertices positions toward their actual positions, especially the vertex indicated by the blue arrow, where the movement was close to five pixels (which is equivalent to 2.5 metres). After the adjustment this distance has been reduced to nearly one pixel. The update of boundary point labels to their corresponding edges in the least squares adjustment is the main contributing factor for this significant improvement. Indeed, this improvement cannot be achieved without the updating process.

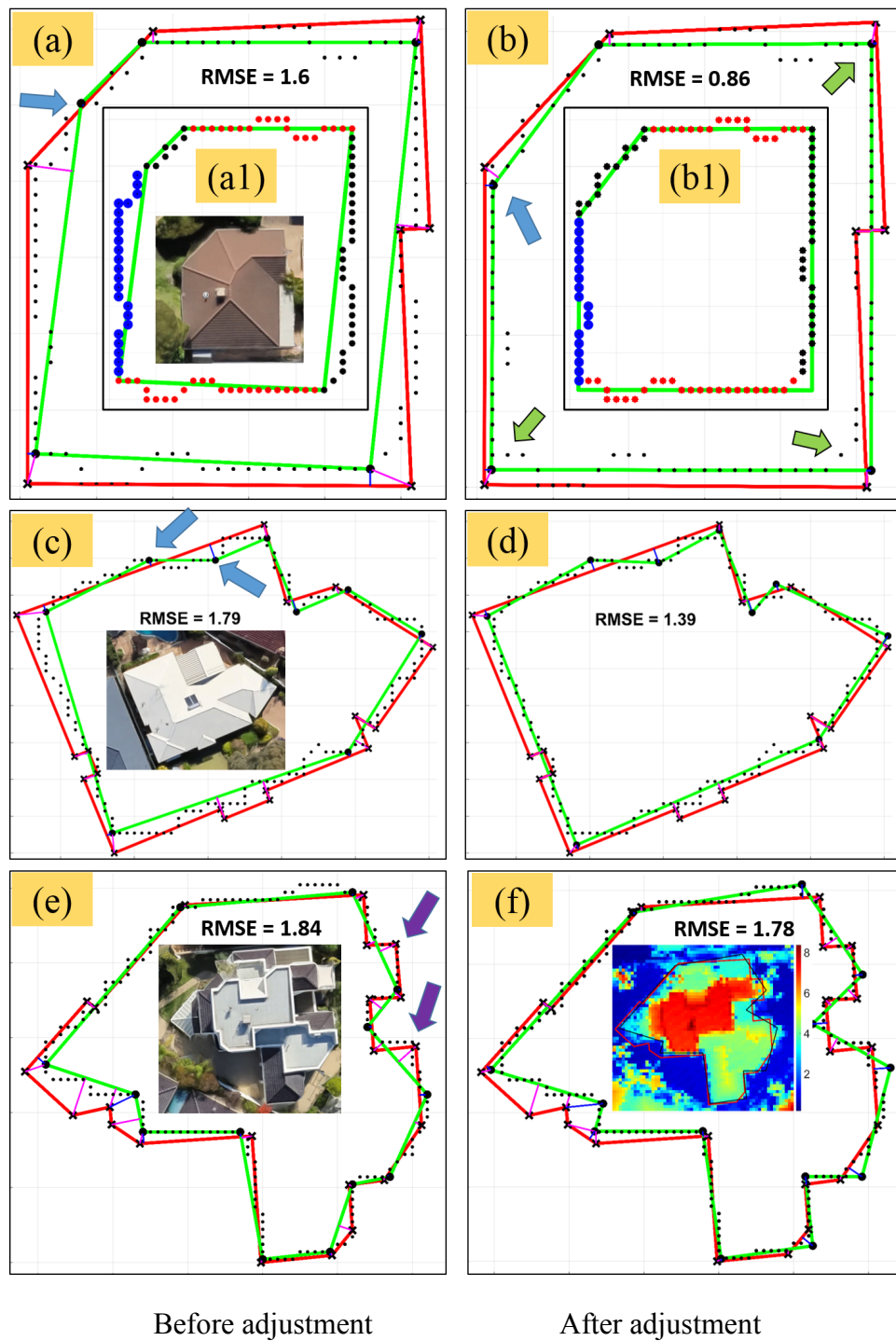


Figure 5.28: Examples show improving planimetric accuracy of non-rectilinear buildings processed by GMM. First column (a), (c) and (f) depict the approximated polygons (green) and the reference (red polygons) before adjustment. (b), (d) and (f) represent the final extracted polygons after adjustment (green) along with references. Images taken from Google Maps are also presented.

For instance, because the GMM adjustment minimises the orthogonal distances between the boundary points and their edge, the labels of points in the blue circle in Figure 5.28 (a1) change from the blue edge to the black edge. In contrast, a further displacement of the vertex indicated by the blue arrow in Figure 5.28 (b) cannot be obtained. Moreover, the resulting best-fit polygon (green lines in Figure 5.28 (b)) achieves a high level of accuracy also because the three right angles indicated by the green arrows have been created correctly.

Another example is presented in Figure 5.28 (c) and (d), in which the improvement percentage is more than 22%. It is highlighted by the blue arrows that redundant vertices are part of the final solution (green polygon) compared to the reference (red polygon). The last row displays a very complex non-rectilinear building Figure 5.28 (e) and (f), where an improvement in planimetric accuracy can be achieved. Nevertheless, the purple arrows in Figure 5.28 (e) indicate that the solution (green outline) is missing vertices compared to the reference (red outline). The reason for the missing vertices is the level of noise in the input data, together with the applied RMSE threshold (RT) in the data-driven algorithm for generating the approximated building polygons. The RT was set to 0.9 m (see Table 5.6). Those missing vertices are able to be detected with more strict RT threshold (e.g. 0.75 m), but consequentially, more redundant vertices will also be obtained, such as those indicated by the blue arrows (Figure 5.28 (c)). Thus, it is a challenge to optimise the RT parameter for noisy data and/or complex of building outlines.

Figure 5.29 shows two non-rectilinear buildings in which the RMSE has been increased slightly after the adjustment. In the first row, the nDSM displays an object (most likely a window overhang) in touch with the building (as highlighted by the black circle), which caused an extra vertex to be created, leading to a higher RMSE value. In contrast, the second row shows an area highlighted by the orange circle in Figure 5.29(d) that has nDSM values less than the nDSM threshold (NT) (two metres, see Table 5.6). Therefore, this area has been lost from the extracted building segment as highlighted by the orange circle in Figure 5.29(e). Thus, the inaccurate boundary points had a negative impact on the quality of the adjusted polygon.

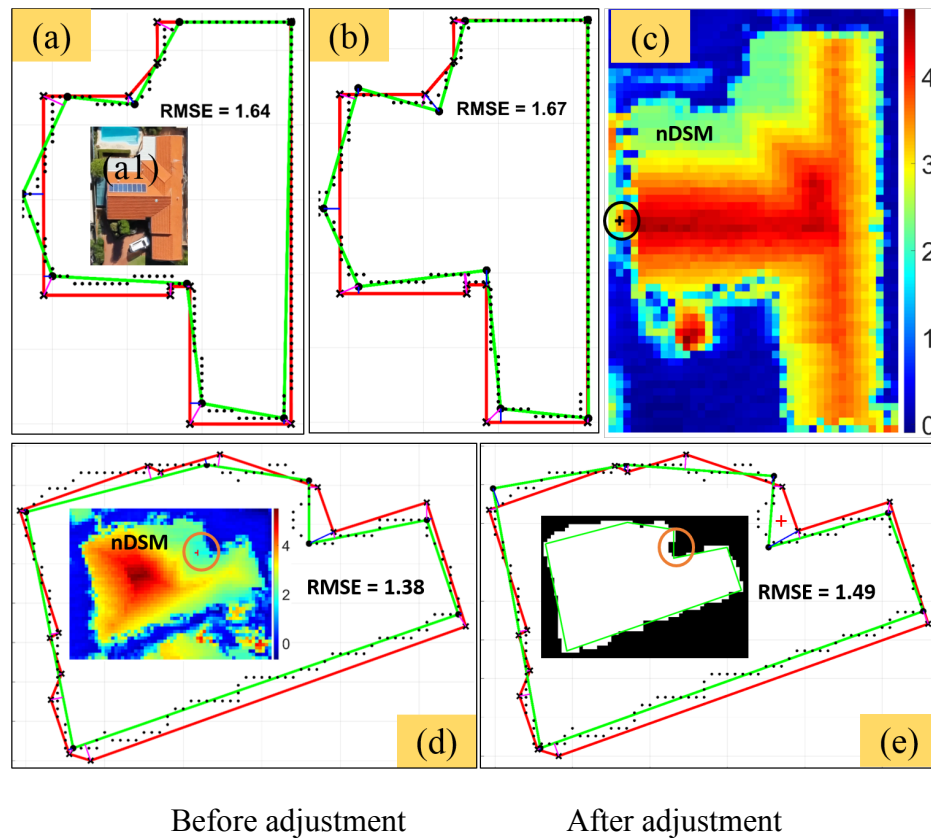


Figure 5.29: Examples show declining planimetric accuracy of two non-rectilinear buildings processed by GMM. Image taken from Google Map (a1), nDSMs, and building segment are also shown.

5.5.4 Assessment of the GHM adjustment

5.5.4.1 Non-Rectilinear buildings processed by GHM adjustment

Table 5.8 shows ten non-rectilinear buildings processed using the GHM adjustment. All of those buildings have a nearly rectilinear shape. Figure 5.30(a) illustrates the calculated RMSE for building polygons before and after the adjustment, while the percentage of the improvement is presented in (b). It can be seen that the planimetric accuracy of 70% for the buildings has been improved by nearly 20%. In contrast, the buildings with decreasing RMSE values is due to over-generalisation. For example,

the GHM adjustment tend to produce a regular shape (rectangular or rectilinear), even though the approximated solution (irregular shape) was closer to the reference.

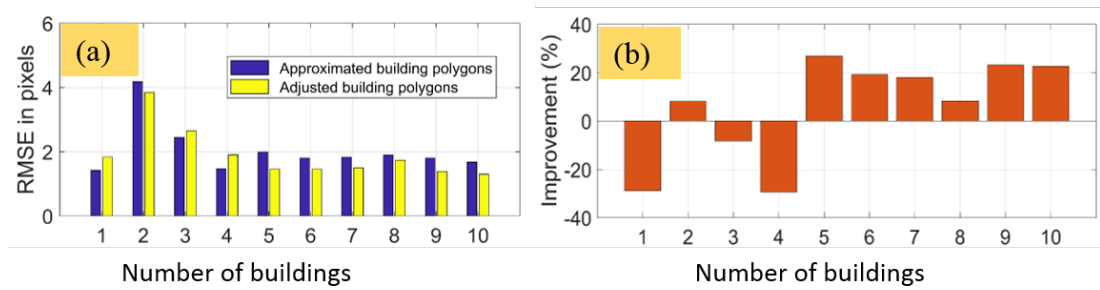


Figure 5.30: Assessment of non-rectilinear building polygons processed using the GHM adjustment. (a) shows the RMSE of the approximated and the adjusted polygons. (b) presents the percentage of the improvement of the RMSE planimetric accuracy.

Figure 5.31 illustrates examples of two non-rectilinear buildings, in which the planimetric accuracy improved by approximately 23% indicating that the GHM

adjustment is able to enhance the geometrical accuracy for some non rectilinear buildings, along with enhancing the regularity.

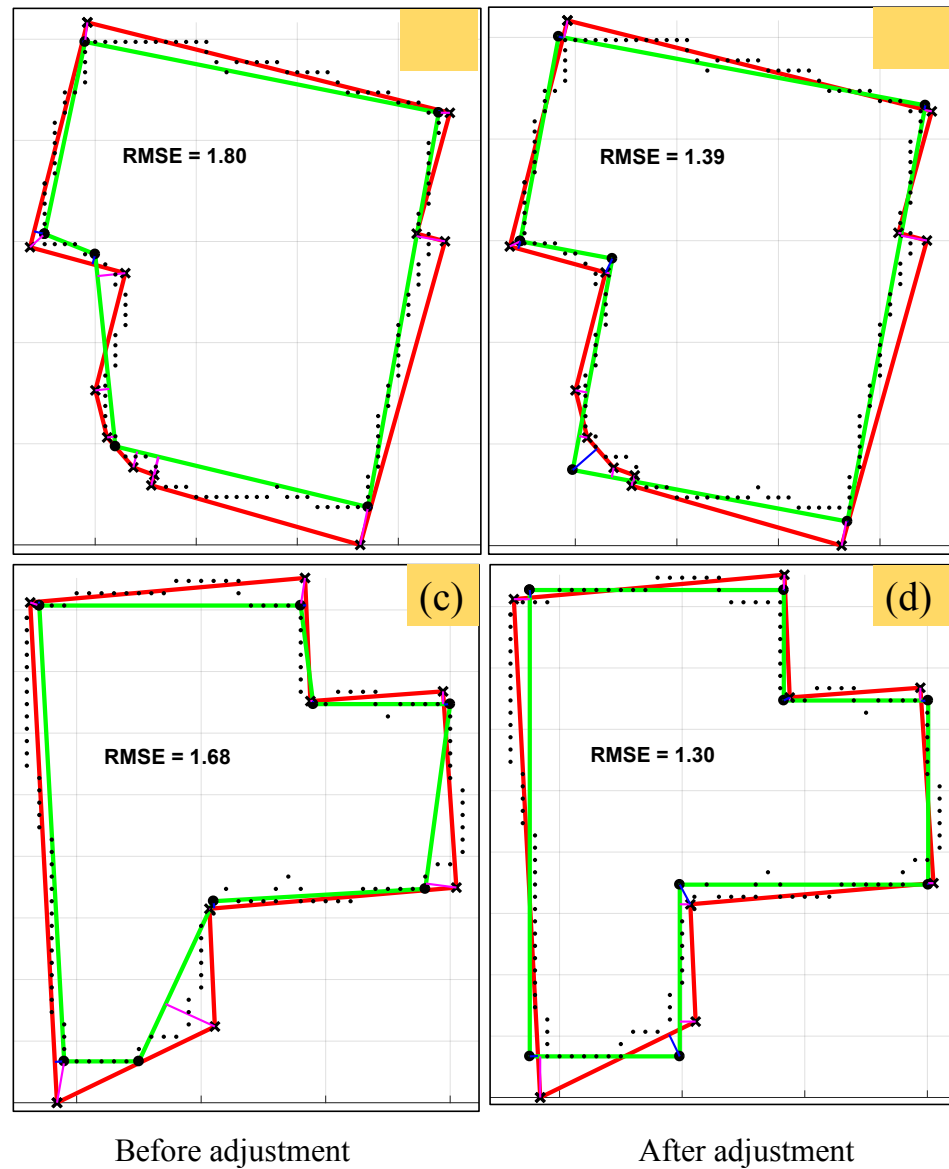


Figure 5.31: Examples show improving planimetric accuracy of two non-rectilinear buildings processed by GHM. (a) and (c) represent the approximated polygons (green) and the reference polygon (red) before adjustment. (b) and (d) represent the final extracted polygons after adjustment (green) along with references polygon (red).

The planimetric accuracy of 3 out of 10 non-rectilinear buildings outlines declined. Figure 5.32 presents the two buildings with the most noticeable increase in RMSE

(approximately 29%). The approximated polygon of the first building has five vertices in Figure 5.32(a) while the final extracted polygon in Figure 5.32(b) shows only four vertices. The approach over-generalised the shape of the building due to the similarity ratio SR of the model-driven method achieving a value of more than 85% (see Table 5.6). Thus, this building has been modelled by a rectangular shape. In contrast, the extracted polygon of second building is more regular in Figure 5.32(d) because the approximated polygon shown in Figure 5.32(c) follows the reference more closely.

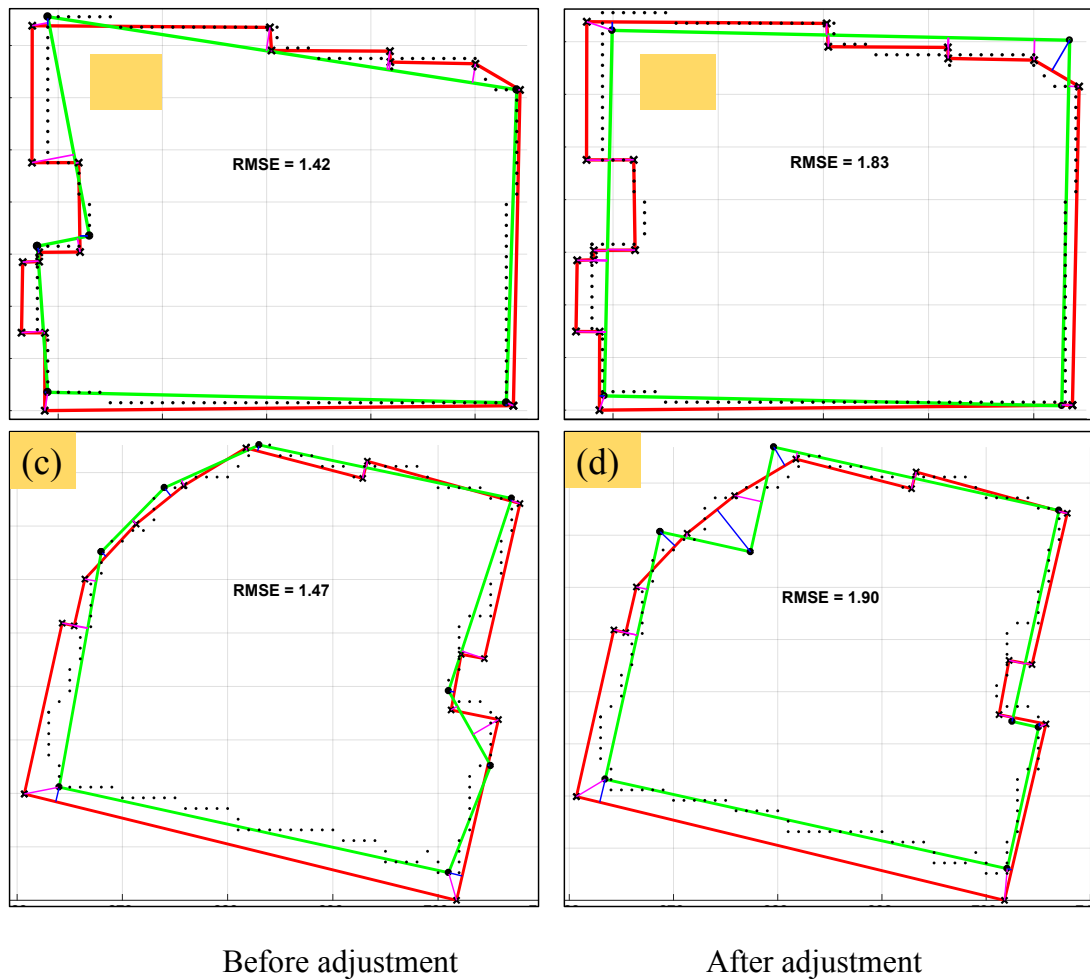


Figure 5.32: Examples with declining planimetric accuracy of non-rectilinear buildings processed using the GHM adjustment. First column (a) and (c) show the approximated polygons (green) and the reference polygons (red) before adjustment. Second column (b) and (d) represent the final extracted polygons after adjustment (green) along with references (red).

5.5.4.2 Rectilinear buildings processed by GHM adjustment

In this subsection, the planimetric accuracy of the 42 rectilinear buildings processed using the GHM adjustment (Table 5.8) is evaluated. The planimetric accuracy of most of the adjusted polygons have significantly improved by nearly 20% compared to the initial approximated polygons as shown Figure 5.33. Some building polygons have been slightly improved, which indicates that the approximated polygons are very close to their actual positions. In contrast, the RMSE of some building polygons after adjustment have decreased significantly, as can be seen for the two buildings presented in Figure 5.34.

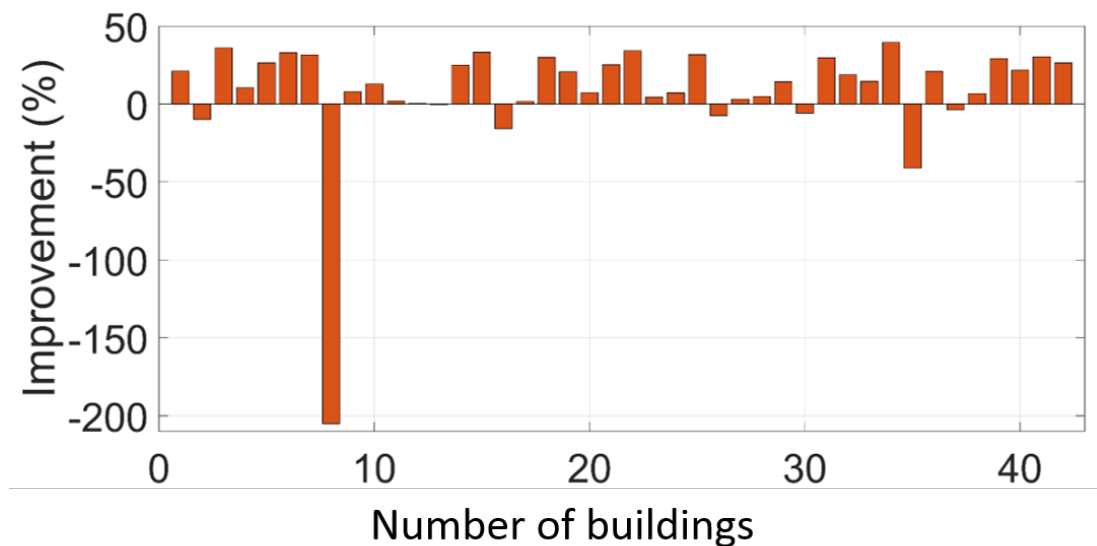


Figure 5.33: Percentage of the improvement of the planimetric accuracy of rectilinear building polygons processed using the GHM adjustment.

The approximated polygon for the building in Figure 5.34 (a) shows two redundant vertices (on the top left and half way down on the right) due to noise. Indeed, the buildings in Figure 5.34(a) and (f) have been modelled as rectangular buildings because their similarity ratio (SR) value was assessed by the applied threshold of 85% (see Table 5.6). It is correct that the approximated polygons (green polygon in Figure 5.34(a)) are closer to the reference (red polygon in Figure 5.34(a)), and therefore the RMSE before the adjustment is smaller than after the adjustment. However, the initial polygon solution of the building (Figure 5.34(a)) has more redundant vertices and a generalisation to a rectangle building (Figure 5.34(b) and (c)) is possible depending on

the user-defined thresholds. Similarly, the adjusted polygon in Figure 5.34(g) (shown in green) has a higher RMSE compared to the approximated polygon in Figure 5.34(f) (also shown in green). Depending on the resolution, the building could also be represented by a rectangle. Images taken from Google Maps are given in Figure 5.34(f1), while the building outline overlaid with the nDSM is given in Figure 5.34(g1).

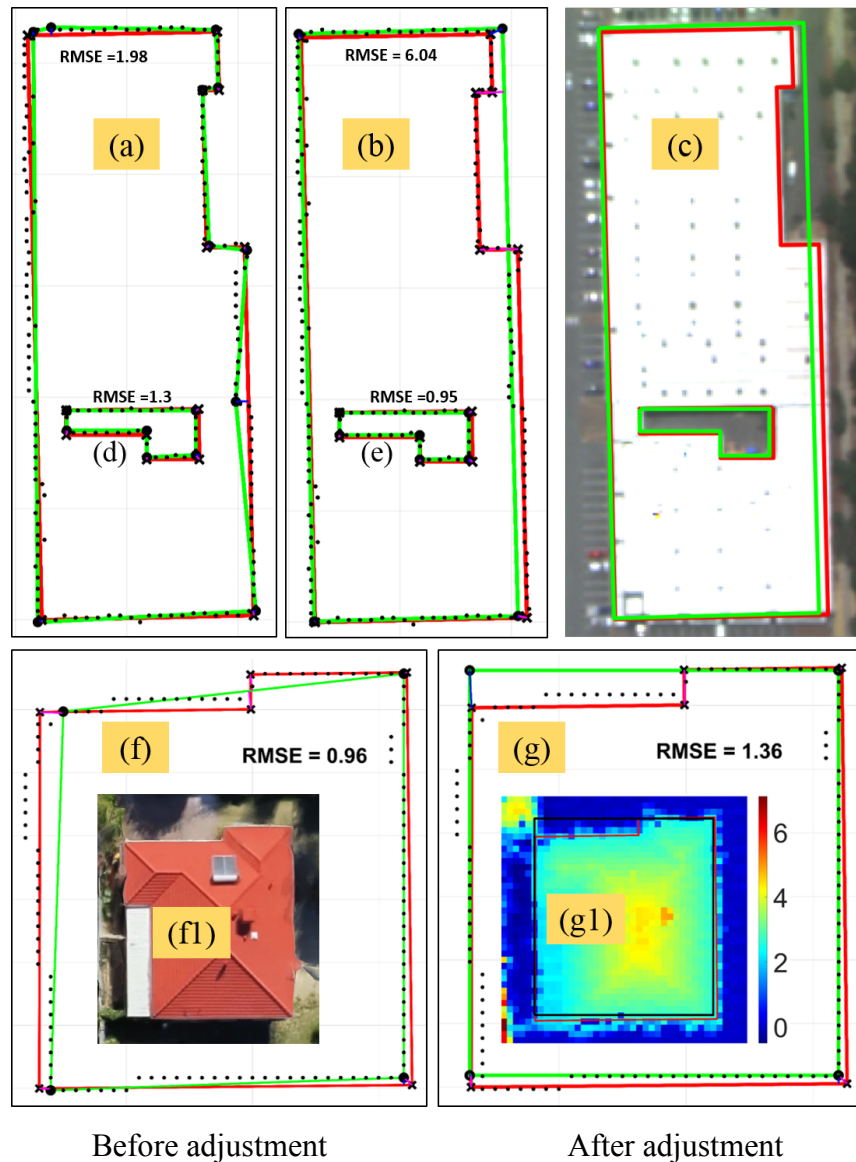


Figure 5.34: Rectilinear buildings processed using the GHM adjustment. The first example showing a building presented in (a), (b), and (c) showing the building before the adjustment, after the adjustment and overlaid with the orthophoto respectively. The reference is always shown as a red polygon. The second example is a building

presented in (f) and (g) before and after the adjustment, respectively. The reference is shown in red. The building polygons are overlaid with images taken from Google Maps (f1) and the nDSM (g1).

Figure 5.35 presents polygons of building outlines where significant improvement of the geometric accuracy have been achieved by the GHM adjustment. More specifically Figure 5.35 shows examples of three rectilinear buildings in which a high level of improvement in terms of best-fit, orientation, and creation of right angles have been obtained after the adjustment compared to the initial approximation. Due to the points representing the extracted building segments in the initial approximation being very noisy (Figure 5.35(a1) and (c1)), the subsequent approximated solutions (green polygons in Figure 5.35(a) and (c)) have some inaccurate vertex positions (blue arrows). Nevertheless, the adjusted polygons (green polygons in Figure 5.35(b) and (d) and (f)) fit the input boundary points and are very similar to the reference polygons. In addition, angles with exactly 90 degrees are always extracted correctly due to the applied constraints in the GHM adjustment. Thus, the final RMSE value of some buildings have scored sub-pixel accuracy level (i.e. Figure 5.35(b) and (f)).

Concerning the dominant building direction, no constraints on the longest polygon edge are required for the proposed method. In contrast, this constraint exists in previous work published by others related to building outline extraction i.e. (Awrangjeb, 2016) and (Höhle, 2017). The problem encountered in work by others is that the longest line segment in a building polygon is not necessarily close enough to the actual building orientation or may be incorrectly estimated. Consequently, this error could contribute to alignment errors in all other line segments in the building polygon and could lead to failing the adjustment. In this thesis, the main direction is estimated from the longest four line-segments in the initial solution. In the subsequent adjustment the dominant building direction is determined automatically after the convergence of iterative least squares adjustment. This is done by a simultaneous minimisation of the orthogonal distances between the polygon edges to their corresponding input boundary points and considering the orthogonality condition in the applied GHM adjustment. Accordingly, the normal components (n_x and n_y) representing the dominant building direction are determined.

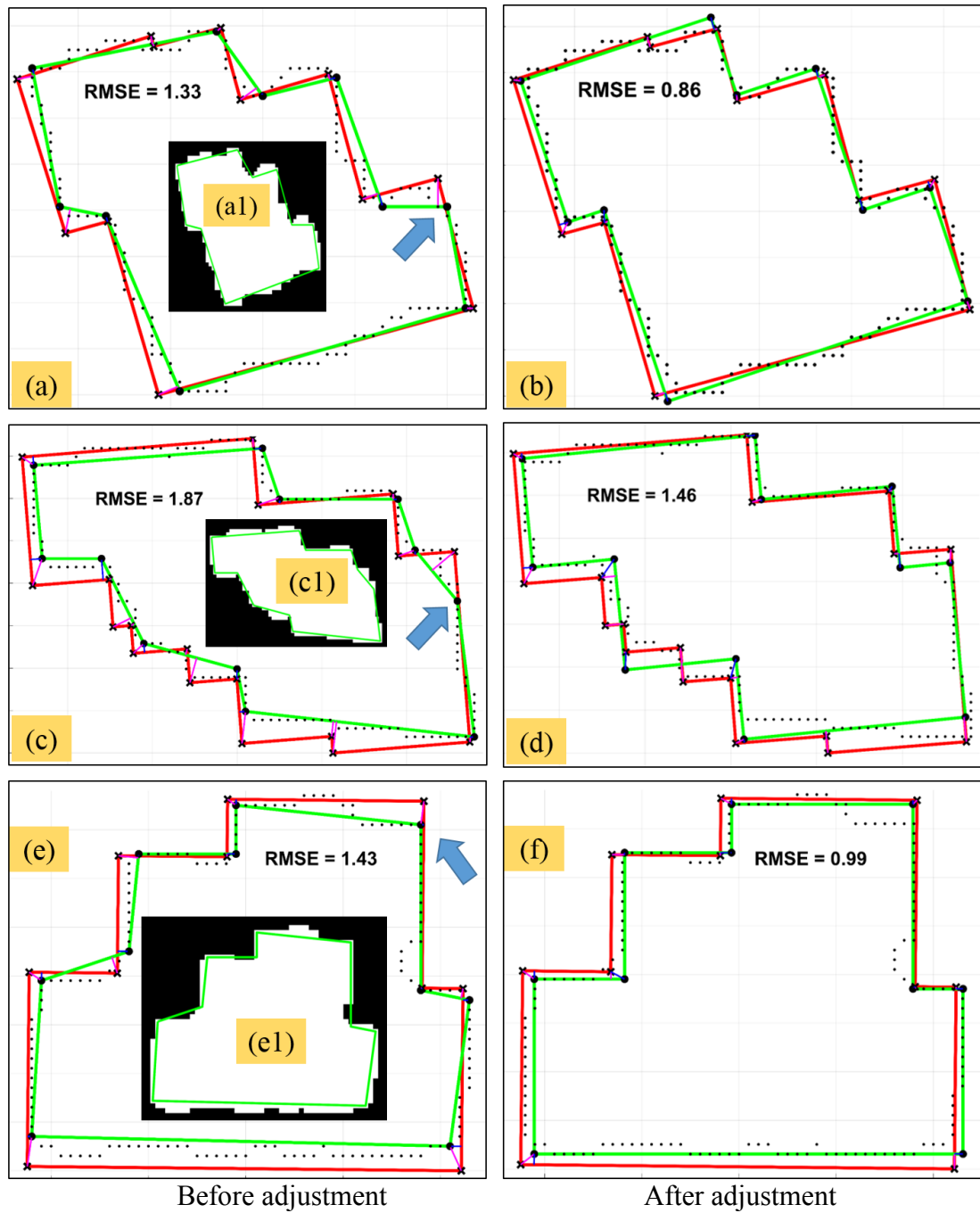


Figure 5.35: Three rectilinear buildings with improved planimetric accuracy processed using the GHM adjustment. The extracted building segments (a), (c),(e) before adjustment; and approximated polygons (b), (d),(f) The reference dataset is shown in red.

6 Conclusions and future work

6.1 Conclusions

The goal of the presented research was to develop an automatic framework for building outline regularisation from 3D laser points while images were used to support the module for removing trees. The newly proposed framework can be summarised as followed: firstly, an accurate DTM is extracted which is then used to calculate an nDSM. In this nDSM building segments are extracted, on which basis polygons representing those buildings are extracted and optimised. Improvement in each of these steps compared to existing approaches have been made and it has been shown that some of their shortcomings can be overcome. These shortcomings have been explained and discussed in the theoretical background and the literature review sections of this thesis.

The first objective of this work was to create an accurate nDSM. This could be achieved by firstly, cleaning the input DSM laser points from outliers followed by transform into a regular DSM. The proposed DTM extraction algorithm is based on selection of seeds or network of ground points as minima, thus denoted as a Network of Ground Points (NGPs). It solves the problem of the local slope structure in a more reliable way and with less complexity using well-distributed scanlines directions. Moreover, the algorithm does not need the slope parameter nor smoothing of the DSM. It relies predominantly on the height difference parameter which turned out to be the most meaningful indicator for DTM creation as well as building detection.

The second objective was to introduce a new fusion method for the robust extraction of building segments. The challenge was the differentiation between building segments and non-building segments such as trees. While cars, low vegetation and vegetation visible in NIR images can be easily removed, and this has been done successfully in previous work, the trees located in shadow regions are challenging. This is because their NDVI values are very low and cannot be used to remove these regions from consideration, without significantly degrading the correctness measure. This limitation was overcome through the successful fusion of the planarity map,

NDVI, and nDSM. Further, it is highlighted that a gridded DSM has the potential of covering missing laser data, at least, partially. Thus, false positive errors are mitigated. The third and major objective of this work was focused on the extraction of polygons representing buildings and their regularisation, especially for complex building structures. This was achieved by combining the robustness of model-driven approaches with the flexibility of data-driven techniques in one comprehensive strategy consisting of three sub-steps:

- A novel data-driven algorithm for creating an accurate approximate building polygon was proposed. In this algorithm, the geometrical characteristics of buildings have been addressed by methodologies such as detection of nearly right angles and area preservation, resulting in a likelihood function. It is proved that this likelihood-based method outperforms the widely used DP algorithm.
- In term of refining rectilinear building polygons, the adjustment procedure has been largely solved. Firstly, the orthogonality constraints of adjacent building edges are integrated into the functional model. Secondly, the best fitting of these edges with their corresponding boundary points is solved by minimising the orthogonal distances between them. In this study, both requirements are simultaneously solved by modifying the Gauss-Helmert Model (GHM) adjustment.
- Estimating an accurate position of building vertices is a challenging task due to several factors such as level of noise in the input boundary points and the complexity of building shapes. To improve the regularisation of the initial building outline solution a method of updating the assignment of boundary points to edges has been introduced. The method is based on measuring and analysing angles and distances between boundary points and building edges and vertices.

Together with aforementioned labelling update method, a new evaluation procedure is proposed considering the RMSE of boundary points relative to building edges as a standard building outline measure. The RMSE is utilised through a combination of the

vertex to vertex distance analysis and vertex to edge RMSE distance analysis. Vertex to Model evaluation (V2M). Without labelling of vertices to their corresponding edges, this calculation of the RMSE cannot be conducted. Because, vertices must be assigned to their corresponding edge in advance, then, distances to closest edge or vertex are measured.

Overall the introduced framework can achieve a satisfactory accuracy level. For instance, the proposed DTM extraction algorithm was demonstrated to out-perform one of the-state-of-the-art method in terms of the complexity, execution time, and quality of the extracted ground regions. For the extracted building segments, as an area-based evaluation, the proposed method showed a high average correctness of 96.23% and an average quality of 88.43% (Table 5.5) in the Vaihingen dataset. In the Melville dataset, the calculated completeness, correctness, and quality for those 125 buildings are 94%, 95%, and 90%, respectively. In term of building outlines regularisation, the achieved RMSE values were 0.91 meter and 1.83 pixels (0.915 meter) in the Vaihingen and the Melville dataset respectively. This sub-meter accuracy level proved the practical efficiency of the proposed workflow associated with a high level of automation.

This study presents a comprehensive workflow for the generation of building outlines in a vector format. The practical efficiency of this proposed workflow on a large scale area has been demonstrated. Further, sub-metre level geometric accuracy of the building polygons has been demonstrated as required for the quality of the generated topographic mapping. The proposed assessment tool is unbiased, free of subjectivity, and which is denoted as Vertex to Model (V2M) evaluation.

This research supports the need for land and urban planners to map building structure, and enables a more automatic approach to map creation and updating over time with a high level of accuracy. Furthermore, the proposed 2D building outlining algorithm can be extended to the application of 3D building model construction. Many of existing 3D modelling assume that 2D plane data are given e.g. (Park & Guldman, 2019) and

(Xiong, Elberink, & Vosselman, 2016), while such data are not always available or they are not up-to-date.

6.2 Future work

Topics for further improvement in 2D building footprint creation can be outlined here. The proposed DTM extraction method was based on three parameters including, the window filter size and two height thresholds for analysing ground points validation. It is recommended that the filter size should be larger than the maximum width of existing buildings in the scene. Therefore, large buildings required a larger filtering size in order to be excluded from the terrain. Consequently, non-terrain and terrain points are difficult to extracted accurately. A possible option to mitigate this limitation is the implementation of adaptive filtering sizes. The filtering size could be adopted based on image analysis results performed on the input image.

In terms of generation of approximate building polygons, the proposed method showed promising results for the estimation of building vertices. However, detecting accurate vertex positions in very complex buildings shapes, especially those having short edges (i.e. one meter or shorter) is still a challenging task. While, the proposed RMSE threshold method for detecting polygon edges is more robust and efficient compared to the DP algorithm, it is still sensitive to noise. Thus, some corners or short edges of buildings can be lost. Although the proposed likelihood equation was applied with fixed weight factors in all tests, it could be further improved through an optimisation of its weight factors.

Additionally, this proposed data-driven method can be made more robust against outliers in combination with the RANSAC procedure. The idea is to categorise boundary points into two groups (inlier and outlier) by measuring their orthogonal distances to the conducted line segment and considering distance threshold. Then, outliers will be removing before calculating the RMSE of the boundary points to be then compare to the user-defined RMSE threshold. This procedure leads to a modelling efficient data-driven algorithm for creating accurate building polygons.

References

- Abed, F. M. A. (2012). Calibration of full-waveform airborne laser scanning data for 3D object segmentation. *Doctoral thesis, School of Civil Engineering and Geosciences, Newcastle University, UK*. 245 pages.,
- Albers, B., Kada, M., & Wichmann, A. (2016). Automatic Extraction and Regularization of Building Outlines from Airborne Lidar Point Clouds. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, XLI-B3*, 555-560. <http://dx.doi.org/10.5194/isprsarchives-XLI-B3-555-2016>
- Alharthy, A., & Bethel, J. (2002). Heuristic filtering and 3D feature extraction from LIDAR data. *International Archives of Photogrammetry Remote Sensing and Spatial Information Sciences*, 34(3/A), 29-34.
- Alsadik, B., & Remondino, F. (2020). Flight Planning for LiDAR-Based UAS Mapping Applications. *ISPRS International Journal of Geo-Information*, 9(6) <http://dx.doi.org/10.3390/ijgi9060378>
- Arefi, H. (2009). *From LiDAR point clouds to 3D building models*. Doctoral thesis, Bundeswehr University Munich, München, Germany. 130 pages
- Avbelj, J. (2015). *Fusion of Hyperspectral Images and Digital Surface Models for Urban Object Extraction*. Doctoral thesis, Technische Universität München, Munich, Germany. 141 pages.
- Avbelj, J., Müller, R., & Bamler, R. (2015). A metric for polygon comparison and building extraction evaluation. *IEEE Geoscience and Remote Sensing Letters*, 12(1), 170-174.
- Awrangjeb, M. (2015). Effective Generation and Update of a Building Map Database Through Automatic Building Change Detection from LiDAR Point Cloud Data. *Remote Sensing*, 7(10), 14119. Retrieved from <http://www.mdpi.com/2072-4292/7/10/14119>
- Awrangjeb, M. (2016). Using point cloud data to identify, trace, and regularize the outlines of buildings. *International Journal of Remote Sensing*, 37(3), 551-579.
- Awrangjeb, M., Lu, G., & Fraser, C. (2014). Automatic building extraction from LiDAR data covering complex urban scenes. *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 40(3), 25.
- Awrangjeb, M., Ravanbakhsh, M., & Fraser, C. S. (2010). Automatic detection of

residential buildings using LIDAR data and multispectral imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, 65(5), 457-467.

Awrangjeb, M., Zhang, C., & Fraser, C. S. (2012). Building detection in complex scenes thorough effective separation of buildings from trees. *Photogrammetric Engineering & Remote Sensing*, 78(7), 729-745.

Axelsson, P. (2000). DEM generation from laser scanner data using adaptive TIN models. *International Archives of Photogrammetry and Remote Sensing*, 33(4), 110-117.

Backes, A. R., & Bruno, O. M. (2013). Polygonal approximation of digital planar curves through vertex betweenness. *Information Sciences*, 222, 795-804. <http://dx.doi.org/10.1016/j.ins.2012.07.062>

Bastanlar, Y., Temizel, A., & Yardimci, Y. (2010). Improved SIFT matching for image pairs with scale difference. *Electronics Letters*, 46(5), 346-348.

Bergsjö, J. (2016). Photogrammetric point cloud generation and surface interpolation for change detection.: (Master thesis. KTH Royal Institute of Technology, Stockholm, Sweden. 59 pages.).

Bethmann, F., & Luhmann, T. (2014). Object-based multi-image semi-global matching—concept and first results. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 40(5), 93-100.

Beumier, C., & Idrissa, M. (2016). Digital terrain models derived from digital surface model uniform regions in urban areas. *International Journal of Remote Sensing*, 37(15), 3477-3493.

Brédif, M., Tournaire, O., Vallet, B., & Champion, N. (2013). Extracting polygonal building footprints from digital surface models: A fully-automatic global optimization framework. *ISPRS journal of photogrammetry and remote sensing*, 77, 57-65.

Breiman, L. (2001). Machine Learning. Vol. 45. *The Netherlands: Kluwer Academic Publishers*, 5-32.

Bulatov, D. (2011). *Textured 3D reconstruction of urban terrain from UAV-borne video sequences.*: Doctoral thesis, Fachrichtung Geodäsie und Geoinformatik der Leibniz-Univ. 152 pages

Bulatov, D., Häufel, G., Meidow, J., Pohl, M., Solbrig, P., & Wernerus, P. (2014). Context-based automatic reconstruction and texturing of 3D urban terrain for quick-response tasks. *ISPRS Journal of Photogrammetry and Remote Sensing*, 93, 157-170.

- Bulatov, D., & Lavery, J. E. (2010). Reconstruction and texturing of 3D urban terrain from uncalibrated monocular images using L1 splines. *Photogrammetric Engineering & Remote Sensing*, 76(4), 439-449.
- Bulatov, D., Wernerus, P., & Heipke, C. (2011). Multi-view dense matching supported by triangular meshes. *ISPRS Journal of Photogrammetry and Remote Sensing*, 66(6), 907-918.
- Canny, J. (1986). A computational approach to edge detection. *IEEE Transactions on pattern analysis and machine intelligence*, (6), 679-698.
- Carrilho, A., Galo, M., & Santos, R. (2018). STATISTICAL OUTLIER DETECTION METHOD FOR AIRBORNE LIDAR DATA. *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences*,
- Chaabouni-Chouayakh, H., Arnau, I. R., & Reinartz, P. (2013). Towards automatic 3-D change detection through multi-spectral and digital elevation model information fusion. *International Journal of Image and Data Fusion*, 4(1), 89-101.
- Chaudhuri, D., & Samal, A. (2007). A simple method for fitting of bounding rectangle to closed regions. *Pattern recognition*, 40(7), 1981-1989.
- Chen, Teo, T.-A., Shao, Y.-C., Lai, Y.-C., & Rau, J.-Y. (2004). Fusion of LIDAR data and optical imagery for building modeling. *International Archives of Photogrammetry and Remote Sensing*, 35(B4), 732-737.
- Chen, L., Zhao, S., Han, W., & Li, Y. (2012). Building detection in an urban area using lidar data and QuickBird imagery. *International Journal of Remote Sensing*, 33(16), 5135-5148.
- Chen, Q., Gong, P., Baldocchi, D., & Xie, G. (2007). Filtering airborne laser scanning data with morphological methods. *Photogrammetric Engineering & Remote Sensing*, 73(2), 175-185.
- Cheng, L., Gong, J., Chen, X., & Han, P. (2008). Building boundary extraction from high resolution imagery and lidar data. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 37(PART B3), 693-698.
- Conte, P., Girelli, V. A., & Mandanici, E. (2018). Structure from Motion for aerial thermal imagery at city scale: Pre-processing, camera calibration, accuracy assessment. *ISPRS journal of photogrammetry and remote sensing*, 146, 320-333.
- Cothren, J. D. (2004). *Reliability in constrained Gauss-Markov models: an analytical and differential approach with applications in photogrammetry* The Ohio State University

- Cramer, M. (2010). The DGPF-test on digital airborne camera evaluation—overview and test design. *Photogrammetrie-Fernerkundung-Geoinformation*, 2010(2), 73-82.
- Demir, N. (2013). *Combination of airborne laser and image data for building detection and 3D roof reconstruction.*: Doctoral thesis, Institute of Geodesy and Photogrammetry, ETH ZURICH. 149 pages.
- Douglas, D. H., & Peucker, T. K. (1973). Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *Cartographica: The International Journal for Geographic Information and Geovisualization*, 10(2), 112-122.
- Du, S., Zhang, Y., Zou, Z., Xu, S., He, X., & Chen, S. (2017). Automatic building extraction from LiDAR data fusion of point and grid-based features. *ISPRS Journal of Photogrammetry and Remote Sensing*, 130, 294-307.
- Dutter, M. (2007). Generalization of building footprints derived from high resolution remote sensing data. *Thesis, Vienna Univ. Technology*. 120 pages.,
- Filin, S., & Pfeifer, N. (2006). Segmentation of airborne laser scanning data using a slope adaptive neighborhood. *ISPRS journal of Photogrammetry and Remote Sensing*, 60(2), 71-80.
- Fischler, M. A., & Bolles, R. C. (1981). Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6), 381-395.
- Förstner, W. (1994). *A framework for low level feature extraction*. Paper presented at the European Conference on Computer Vision.
- Förstner, W., & Wrobel, B. P. (2016). *Photogrammetric computer vision- Statistics, Geometry, Orientation and Reconstruction.*: Geometry.
- Geibel, R., & Stilla, U. (2000). Segmentation of laser altimeter data for building reconstruction: different procedures and comparison. *International Archives of Photogrammetry and Remote Sensing*, 33(B3/1; PART 3), 326-334.
- Gerke, M., Heipke, C., & Straub, B.-M. (2001). *Building extraction from aerial imagery using a generic scene model and invariant geometric moments*. Paper presented at the Remote Sensing and Data Fusion over Urban Areas, IEEE/ISPRS Joint Workshop 2001
- Gerke, M., & Xiao, J. (2014). Fusion of airborne laserscanning point clouds and images for supervised and unsupervised scene classification. *ISPRS Journal of Photogrammetry and Remote Sensing*, 87, 78-92.

- Gevaert, C., Persello, C., Nex, F., & Vosselman, G. (2018). A deep learning approach to DTM extraction from imagery using rule-based training labels. *ISPRS journal of photogrammetry and remote sensing*, 142, 106-123.
- Gonzalez, R. C., Eddins, S. L., & Woods, R. E. (2004). *Digital Image Publishing Using MATLAB*: Prentice Hall.
- Griffiths, D., & Boehm, J. (2019). Improving public data for building segmentation from Convolutional Neural Networks (CNNs) for fused airborne lidar and image data using active contours. *ISPRS Journal of Photogrammetry and Remote Sensing*, 154, 70-83.
- Gross, H., & Thoennessen, U. (2006). *Extraction of lines from laser point clouds*. Paper presented at the Symposium of ISPRS Commission III: Photogrammetric Computer Vision PCV06. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences
- Grubbs, F. E. (1969). Procedures for detecting outlying observations in samples. *Technometrics*, 11(1), 1-21.
- Guercke, R., & Sester, M. (2011). *Building footprint simplification based on hough transform and least squares adjustment*. Paper presented at the Proceedings of the 14th workshop of the ICA commission on generalisation and multiple representation, Paris
- Haala, N., & Kada, M. (2010). An update on automatic 3D building reconstruction. *ISPRS Journal of Photogrammetry and Remote Sensing*, 65(6), 570-580.
- Habib, A., Ghanma, M., & Mitishita, E. (2004). Co-registration of Photogrammetric and Lidar Data: Methodology and case study. *Revista Brasileira de Cartografia*, 1(56)
- Hammoudi, K., & Dornaika, F. (2010). A featureless approach to 3D polyhedral building modeling from aerial images. *Sensors*, 11(1), 228-259.
- Haralick, R. M., Sternberg, S. R., & Zhuang, X. (1987). Image analysis using mathematical morphology. *IEEE transactions on pattern analysis and machine intelligence*, (4), 532-550.
- Hartley, R., & Zisserman, A. (2000). Multiple View Geometry in Computer Vision, Cambridge Uni. Pr., Cambridge, UK,
- Hastedt, H., & Luhmann, T. (2015). Investigations on the quality of the interior orientation and its impact in object space for UAV photogrammetry. *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences*, 40
- He, Y., Zhang, C., & Fraser, C. S. (2014). An energy minimization approach to

automated extraction of regular building footprints from airborne LiDAR data. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2(3), 65.

Helmert, F. R. (1872). Die Ausgleichsrechnung nach der Methode der kleinsten Quadrate.

Henn, A., Gröger, G., Stroh, V., & Plümer, L. (2013). Model driven reconstruction of roofs from sparse LIDAR point clouds. *ISPRS Journal of photogrammetry and remote sensing*, 76, 17-29.

Hingee, K., Caccetta, P., Caccetta, L., Wu, X., & Devereaux, D. (2016). Digital terrain from a two-step segmentation and outlier-based algorithm. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 3, 233-239.

Hirschmuller, H. (2008). Stereo processing by semiglobal matching and mutual information. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 30(2), 328-341. <http://dx.doi.org/10.1109/TPAMI.2007.1166>

Hodgson, M. E., & Bresnahan, P. (2004). Accuracy of airborne LiDAR-derived elevation. *Photogrammetric Engineering & Remote Sensing*, 70(3), 331-339.

Höhle, J. (2017). Generating Topographic Map Data from Classification Results. *Remote Sensing*, 9(3), 224.

Hong, G., & Zhang, Y. (2007). *Combination of feature-based and area-based image registration technique for high resolution remote sensing image*. Paper presented at the 2007 IEEE International Geoscience and Remote Sensing Symposium

Hough, P. V. (1962). Method and means for recognizing complex patterns. (Google Patents).

Hu, X., & Yuan, Y. (2016). Deep-Learning-Based Classification for DTM Extraction from ALS Point Cloud. *Remote Sensing*, 8(9), 730. Retrieved from <https://www.mdpi.com/2072-4292/8/9/730>

Huang, X., & Zhang, L. (2011). A multidirectional and multiscale morphological index for automatic building extraction from multispectral GeoEye-1 imagery. *Photogrammetric Engineering & Remote Sensing*, 77(7), 721-732.

Hui, Z., Jin, S., Cheng, P., Ziggah, Y. Y., Wang, L., Wang, Y., . . . Hu, Y. (2019). An Active Learning Method for DEM Extraction From Airborne LiDAR Point Clouds. *IEEE Access*, 7, 89366-89378.

Jarząbek-Rychard, M. (2012). Reconstruction of building outlines in dense urban areas

based on LiDAR data and address points. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 39, B3.

Jwa, Y., Sohn, G., Tao, V., & Cho, W. (2008). An implicit geometric regularization of 3d building shape using airborne lidar data. *Inter. Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 37, 69-76.

Kada, M., & McKinley, L. (2009). 3D building reconstruction from LiDAR based on a cell decomposition approach. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 38(Part 3), W4.

Kilian, J., Haala, N., & Englich, M. (1996). Capture and evaluation of airborne laser scanner data. *International Archives of Photogrammetry and Remote Sensing*, 31, 383-388.

Koch, K.-R. (1999). *Parameter estimation and hypothesis testing in linear models*: Springer Science & Business Media.

Kraus, K., & Pfeifer, N. (1998). Determination of terrain models in wooded areas with airborne laser scanner data. *ISPRS Journal of Photogrammetry and Remote Sensing*, 53(4), 193-203.

Kwak, E. (2013). *Automatic 3D building model generation by integrating LiDAR and aerial images using a hybrid approach*. Doctoral thesis, University of Calgary, Canada. 180 pages.

Kwak, E., & Habib, A. (2014). Automatic representation and reconstruction of DBM from LiDAR data using Recursive Minimum Bounding Rectangle. *ISPRS Journal of Photogrammetry and Remote Sensing*, 93, 171-191.

Lafarge, F., Descombes, X., Zerubia, J., & Pierrot-Deseilligny, M. (2008). Automatic building extraction from DEMs using an object approach and application to the 3D-city modeling. *ISPRS Journal of Photogrammetry and Remote Sensing*, 63(3), 365-381.

Lari, Z., Habib, A., & Kwak, E. (2011). *An adaptive approach for segmentation of 3D laser point cloud*. Paper presented at the ISPRS Workshop laser scanning

Leberl, F., Irschara, A., Pock, T., Meixner, P., Gruber, M., Scholz, S., & Wiechert, A. (2010). Point clouds. *Photogrammetric Engineering & Remote Sensing*, 76(10), 1123-1134.

Lee, H. S., & Younan, N. H. (2003). DTM extraction of LiDAR returns via adaptive processing. *IEEE Transactions on Geoscience and Remote Sensing*, 41(9), 2063-2069.

- Li, M., Rottensteiner, F., & Heipke, C. (2019). Modelling of buildings from aerial LiDAR point clouds using TINs and label maps. *ISPRS Journal of Photogrammetry and Remote Sensing*, 154, 127-138.
- Liu, X. (2008). Airborne LiDAR for DEM generation: some critical issues. *Progress in Physical Geography*, 32(1), 31-49.
- Long, J., Shelhamer, E., & Darrell, T. (2015). *Fully convolutional networks for semantic segmentation*. Paper presented at the Proceedings of the IEEE conference on computer vision and pattern recognition
- Lowe, D. G. (1999). *Object recognition from local scale-invariant features*. Paper presented at the iccv
- Ma, R. (2004). *Building model reconstruction from LiDAR data and aerial photographs*. Doctoral thesis, The Ohio State University, USA. 181 pages
- Maas, H.-G., & Vosselman, G. (1999). Two algorithms for extracting building models from raw laser altimetry data. *ISPRS Journal of photogrammetry and remote sensing*, 54(2), 153-163.
- Marcos, D., Volpi, M., Kellenberger, B., & Tuia, D. (2018). Land cover mapping at very high resolution with rotation equivariant CNNs: Towards small yet accurate models. *ISPRS Journal of Photogrammetry and Remote Sensing*,
- Marmanis, D., Schindler, K., Wegner, J. D., Galliani, S., Datcu, M., & Stilla, U. (2018). Classification with an edge: Improving semantic image segmentation with boundary detection. *ISPRS Journal of Photogrammetry and Remote Sensing*, 135, 158-172.
- McAndrew, A. (2004). An introduction to digital image processing with matlab notes for scm2511 image processing. *School of Computer Science and Mathematics, Victoria University of Technology*, 264(1)
- Meng, X., Currit, N., & Zhao, K. (2010). Ground filtering algorithms for airborne LiDAR data: A review of critical issues. *Remote Sensing*, 2(3), 833-860.
- Meng, X., Wang, L., & Currit, N. (2009). Morphology-based building detection from airborne LIDAR data. *Photogrammetric Engineering & Remote Sensing*, 75(4), 437-442.
- Meng, X., Wang, L., Silván-Cárdenas, J. L., & Currit, N. (2009). A multi-directional ground filtering algorithm for airborne LIDAR. *ISPRS Journal of Photogrammetry and Remote Sensing*, 64(1), 117-124.

- MicaSense. (2017). RedEdge Specifications. <https://support.micasense.com/hc/en-us/articles/115003537673-RedEdge-M-User-Manual-PDF->.
- Mikhail, E. M., & Ackermann, F. E. (1976). Observations and least squares. *IEP, New York, NY, USA.*,
- Mikolajczyk, K., & Schmid, C. (2005). A performance evaluation of local descriptors.
- Misra, P., Avtar, R., & Takeuchi, W. (2018). Comparison of Digital Building Height Models Extracted from AW3D, TanDEM-X, ASTER, and SRTM Digital Surface Models over Yangon City. *Remote Sensing*, 10(12), 2008.
- Mongus, D., Lukač, N., & Žalik, B. (2014). Ground and building extraction from LiDAR data based on differential morphological profiles and locally fitted surfaces. *ISPRS Journal of Photogrammetry and Remote Sensing*, 93, 145-156.
- Mousa, Y., Helmholz, P., & Belton, D. (2017). NEW DTM EXTRACTION APPROACH FROM AIRBORNE IMAGES DERIVED DSM. *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.*, 42
- Mousa, Y. A., Helmholz, P., Belton, D., & Bulatov, D. (2019). Building detection and regularisation using DSM and imagery information. *The Photogrammetric Record*, 34(165), 85-107. <http://dx.doi.org/10.1111/phor.12275>
- Müller, S., & Zaum, D. W. (2005). Robust building detection in aerial images. *International Archives of Photogrammetry and Remote Sensing*, 36(B2/W24), 143-148.
- Neidhart, H., & Sester, M. (2008). Extraction of building ground plans from LiDAR data. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 37(PART B2)
- Olson, C. F., & Huttenlocher, D. P. (1997). *Automatic target recognition by matching oriented edge pixels.*
- Ozcan, A., Unsalan, C., & Reinartz, P. (2013). *Building detection using local features and DSM data.* Paper presented at the Recent Advances in Space Technologies (RAST), 2013 6th International Conference on
- Park, Y., & Guldmann, J.-M. (2019). Creating 3D city models with building footprints and LIDAR point cloud classification: A machine learning approach. *Computers, Environment and Urban Systems*, 75, 76-89.
- Partovi, T., Bahmanyar, R., Krauß, T., & Reinartz, P. (2017). Building Outline Extraction Using a Heuristic Approach Based on Generalization of Line Segments.

IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 10(3), 933-947. <http://dx.doi.org/10.1109/JSTARS.2016.2611861>

Passini, R., & Jacobsen, K. (2004). Accuracy analysis of digital orthophotos from very high resolution imagery. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences-ISPRS Archives 35, Part B4 (2004)*, 35, 695-700.

Pérez-García, J., Delgado, J., Cardenal, J., Colomo, C., & Ureña, M. (2012). Progressive densification and region growing methods for LIDAR data classification. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 39(B3), 155-160.

Perko, R., Raggam, H., Gutjahr, K., & Schardt, M. (2015). Advanced DTM generation from very high resolution satellite stereo images. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 1, pp. 165-172.

Pfeifer, N., Rutzinger, M., Rottensteiner, F., Muecke, W., & Hollaus, M. (2007). *Extraction of building footprints from airborne laser scanning: Comparison and validation techniques*. Paper presented at the Urban Remote Sensing Joint Event, 2007

Piltz, B., Bayer, S., & Poznanska, A. M. (2016). Volume Based Dtm Generation from Very High Resolution Photogrammetric Dsms. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, XLI-B3*, 83-90. <http://dx.doi.org/10.5194/isprsarchives-XLI-B3-83-2016>

Piramanayagam, S., Saber, E., Schwartzkopf, W., & Koehler, F. (2018). Supervised Classification of Multisensor Remotely Sensed Images Using a Deep Learning Framework. *Remote Sensing*, 10(9), 1429.

Pohl, M., & Feldmann, D. (2016). Generating Straight Outlines of 2D Point Sets and Holes using Dominant Directions or Orthogonal Projections *Proceedings of the 11th Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications: Volume 1: GRAPP, held in Rome, Italy*, (pp. 59-71). 3021610: SCITEPRESS - Science and Technology Publications, Lda. <http://dx.doi.org/10.5220/0005720300570069>

Pohl, M., Meidow, J., & Bulatov, D. (2017). *Simplification of Polygonal Chains by Enforcing Few Distinctive Edge Directions*. Paper presented at the Scandinavian Conference on Image Analysis

Potůčková, M., & Hofman, P. (2016). Comparison of Quality Measures for Building Outline Extraction. *The Photogrammetric Record*, 31(154), 193-209.

Qin, R., & Fang, W. (2014). A hierarchical building detection method for very high

resolution remotely sensed images combined with DSM using graph cut optimization. *Photogrammetric Engineering & Remote Sensing*, 80(9), 873-883.

Riegl. (2017). Riegl VZ-1000. https://www.3dlasermapping.com/wp-content/uploads/2017/10/DataSheet_VZ-2000_2017-06-07.pdf.

Rottensteiner, F., Sohn, G., Gerke, M., Wegner, J. D., Breitkopf, U., & Jung, J. (2014). Results of the ISPRS benchmark on urban object detection and 3D building reconstruction. *ISPRS Journal of Photogrammetry and Remote Sensing*, 93, 256-271.

Rottensteiner, F., Trinder, J., Clode, S., & Kubik, K. (2005). Using the Dempster–Shafer method for the fusion of LIDAR data and multi-spectral images for building detection. *Information fusion*, 6(4), 283-300.

Rottensteiner, F., Trinder, J., Clode, S., & Kubik, K. (2007). Building detection by fusion of airborne laser scanner data and multi-spectral images: Performance evaluation and sensitivity analysis. *ISPRS Journal of Photogrammetry and Remote Sensing*, 62(2), 135-149.

Rutzinger, M., Rottensteiner, F., & Pfeifer, N. (2009). A comparison of evaluation techniques for building extraction from airborne laser scanning. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 2(1), 11-20.

Sampath, A., & Shan, J. (2007). Building boundary tracing and regularization from airborne LiDAR point clouds. *Photogrammetric Engineering & Remote Sensing*, 73(7), 805-812.

Sester, M., & Neidhart, H. (2008). Reconstruction of building ground plans from laser scanner data. *AGILE08*, 111.

Shan, J., & Sampath, A. (2005). Urban DEM generation from raw LiDAR data. *Photogrammetric Engineering & Remote Sensing*, 71(2), 217-226.

Shi, W., & Cheung, C. (2006). Performance evaluation of line simplification algorithms for vector generalization. *The Cartographic Journal*, 43(1), 27-44.

Sirmacek, B., d'Angelo, P., & Reinartz, P. (2010). *Detecting complex building shapes in panchromatic satellite images for digital elevation model enhancement*. Paper presented at the ISPRS Workshop on Modeling of Optical Airborne and Space-borne Sensors, Istanbul, Turkey

Sithole, G. (2001). Filtering of laser altimetry data using a slope adaptive filter. *International Archives of Photogrammetry Remote Sensing and Spatial Information Sciences*, 34(3/W4), 203-210.

- Sithole, G., & Vosselman, G. (2004). Experimental comparison of filter algorithms for bare-Earth extraction from airborne laser scanning point clouds. *ISPRS journal of photogrammetry and remote sensing*, 59(1), 85-101.
- Sithole, G., & Vosselman, G. (2005). Filtering of airborne laser scanner data based on segmented point clouds. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 36(part 3), W19.
- Skaloud, J., & Lichti, D. (2006). Rigorous approach to bore-sight self-calibration in airborne laser scanning. *ISPRS Journal of Photogrammetry and Remote Sensing*, 61(1), 47-59. <http://dx.doi.org/10.1016/j.isprsjprs.2006.07.003>
- Sohn, G., & Dowman, I. (2007). Data fusion of high-resolution satellite imagery and LiDAR data for automatic building extraction. *ISPRS Journal of Photogrammetry and Remote Sensing*, 62(1), 43-63.
- Sohn, G., Jwa, Y., Jung, J., & Kim, H. (2012). An implicit regularization for 3D building rooftop modeling using airborne LIDAR Data. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 1(3), 305-310.
- Song, J., & Miao, R. (2016). A novel evaluation approach for line simplification algorithms towards vector map visualization. *ISPRS International Journal of Geo-Information*, 5(12), 223.
- Stucker, C., Richard, A., Wegner, J. D., & Schindler, K. (2018). Supervised outlier detection in large-scale MVS point clouds for 3D city modeling applications. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 4(2), 263-270.
- Tarsha-Kurdi, F., Landes, T., Grussenmeyer, P., & Koehl, M. (2007). *Model-driven and data-driven approaches using LIDAR data: Analysis and comparison*. Paper presented at the ISPRS Workshop, Photogrammetric Image Analysis (PIA07)
- Tian, J., Chaabouni-Chouayakh, H., & Reinartz, P. (2011). *3D building change detection from high resolution spaceborne stereo imagery*. Paper presented at the Multi-Platform/Multi-Sensor Remote Sensing and Mapping (M2RSM), 2011 International Workshop on
- Tian, J., & Reinartz, P. (2013). *Fusion of multi-spectral bands and DSM from Worldview-2 Stereo imagery for building extraction*. Paper presented at the Urban Remote Sensing Event (JURSE), 2013 Joint
- Toldo, R., & Fusiello, A. (2008). *Robust multiple structures estimation with j-linkage*. Paper presented at the European conference on computer vision

- Tóvári, D., & Pfeifer, N. (2005). Segmentation based robust interpolation-a new approach to laser data filtering. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 36(3/19), 79-84.
- Vallet, B., Pierrot-Deseilligny, M., Boldo, D., & Brédif, M. (2011). Building footprint database improvement for 3D reconstruction: A split and merge approach and its evaluation. *ISPRS journal of photogrammetry and remote sensing*, 66(5), 732-742.
- Vosselman, G. (2000). Slope based filtering of laser altimetry data. *International Archives of Photogrammetry and Remote Sensing*, 33(B3/2; PART 3), 935-942.
- Vosselman, G., Gorte, B. G., Sithole, G., & Rabbani, T. (2004). Recognising structure in laser scanner point clouds. *International archives of photogrammetry, remote sensing and spatial information sciences*, 46(8), 33-38.
- Vosselman, G., Kessels, P., & Gorte, B. (2005). The utilisation of airborne laser scanning for mapping. *International Journal of Applied Earth Observation and Geoinformation*, 6(3-4), 177-186.
- Vosselman, G., & Maas, H.-G. (2010). *Airborne and terrestrial laser scanning*.: CRC: Boca Raton, FL, USA. p. 318.
- Wack, R., & Wimmer, A. (2002). Digital terrain models from airborne laserscanner data-a grid based approach. *International Archives of Photogrammetry Remote Sensing and Spatial Information Sciences*, 34(3/B), 293-296.
- Wang, O., Lodha, S. K., & Helmbold, D. P. (2006). *A bayesian approach to building footprint extraction from aerial lidar data*. Paper presented at the 3D Data Processing, Visualization, and Transmission, Third International Symposium on
- Weinmann, M., Urban, S., Hinz, S., Jutzi, B., & Mallet, C. (2015). Distinctive 2D and 3D features for automated large-scale scene analysis in urban areas. *Computers & Graphics*, 49, 47-57.
- West, K. F., Webb, B. N., Lersch, J. R., Pothier, S., Triscari, J. M., & Iverson, A. E. (2004). *Context-driven automated target detection in 3D data*. Paper presented at the Automatic Target Recognition XIV
- Widyaningrum, E., Gorte, B., & Lindenbergh, R. (2019). Automatic Building Outline Extraction from ALS Point Clouds by Ordered Points Aided Hough Transform. *Remote Sensing*, 11(14), 1727. Retrieved from <https://www.mdpi.com/2072-4292/11/14/1727>
- Xiong, B., Elberink, S. O., & Vosselman, G. (2016). FOOTPRINT MAP PARTITIONING USING AIRBORNE LASER SCANNING DATA. *ISPRS Annals*

of Photogrammetry, Remote Sensing & Spatial Information Sciences, 3(3)

Yan, M., Blaschke, T., Liu, Y., & Wu, L. (2012). An object-based analysis filtering algorithm for airborne laser scanning. *International journal of remote sensing*, 33(22), 7099-7116.

Yoon, J.-S., Shin, J.-I., & Lee, K.-S. (2008). Land cover characteristics of airborne LiDAR intensity data: A case study. *IEEE Geoscience and Remote Sensing Letters*, 5(4), 801-805.

Zhang, K., Chen, S.-C., Whitman, D., Shyu, M.-L., Yan, J., & Zhang, C. (2003). A progressive morphological filter for removing nonground measurements from airborne LIDAR data. *IEEE transactions on geoscience and remote sensing*, 41(4), 872-882.

Zhang, K., & Whitman, D. (2005). Comparison of three algorithms for filtering airborne lidar data. *Photogrammetric Engineering & Remote Sensing*, 71(3), 313-324.

Zhang, K., Yan, J., & Chen, S.-C. (2006). Automatic construction of building footprints from airborne LIDAR data. *IEEE Transactions on Geoscience and Remote Sensing*, 44(9), 2523-2533.

Zhao, Z., Duan, Y., Zhang, Y., & Cao, R. (2016). Extracting buildings from and regularizing boundaries in airborne lidar data using connected operators. *International Journal of Remote Sensing*, 37(4), 889-912.

Zhao, Z., & Saalfeld, A. (1997). *Linear-time sleeve-fitting polyline simplification algorithms*. Paper presented at the Proceedings of AutoCarto